TRW Defense &
Space System Group
One Space Park
Redondo Beach, CA 90278

Final Report

Space Power Distribution System Technology

Volume 2 Autonomous Power Management

TRW Report Number 34579-6001-UT-00

March 1983

Contract NAS8-33198

Prepared for
George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Marshall Space Flight Center, Alabama 35812

TRW Defense &
Space System Group
One Space Park
Redondo Beach, CA 90278

TRW

Final Report

Space Power Distribution System Technology

Volume 2 Autonomous Power Management

TRW Report Number 34579-6001-UT-00

March 1983

Contract NAS8-33198

Prepared for
George C. Marshall Space Flight Center
National Aeronautics and Space Administration
Marshall Space Flight Center, Alabama 35812

FOREWORD

This report documents in three volumes the work performed by TRW Electronics and Defense Sector, Redondo Beach, California, for George C. Marshall Space Flight Center (NASA/MSFC), Huntsville, Alabama, under contract NAS8-33198 (TRW Sales Number 34579).

Volume 1, "Reference EPS Design," summarizes the work under Task 1, System Design and Technology Development; Volume 2, Autonomous Power Management, summarizes the work under Task 2, Power Management Subsystem Development; and Volume 3, Test Facility Design, summarizes the work under Task 3, AMPS Test Facility. This final report is submitted in compliance with the contract statement of work and covers the entire period of performance from 1978 December 05 through 1982 March 31.

These three tasks were structured to define, develop, and demonstrate technology for autonomous management of complex multi-hundred-kilowatt electrical power subsystems for orbital spacecraft. Initially, a conceptual design of a reference electrical power subsystem was developed from spacecraft level life cycle cost analyses of 1985-86 technology for solar array, energy storage, power distribution, shuttle transportation, and orbital drag makeup propulsion (Volume 1). This reference electrical power subsystem was subsequently utilized to quantify the benefits of the power management approach and to demonstrate the power management subsystem concept (Volume 2). It is important to recognize that the resultant power management technology (strategies and hardware) has application to a broad spectrum of electrical power systems and is independent of power level, distribution voltage and form (ac or dc), payload type, spacecraft mission, and orbital parameters.

This study was managed for TRW by Charles Sollo of the Electrical Power Systems Laboratory, and for NASA/MSFC by Jim Graves of the Power Branch. The principal contributors for this technical study task and preparation of this report volume include D. Kent Decker, Marshall D. Cannady, John E. Cassinelli, Bertrand F. Farber, Charles Lurie, Gerald W. Fleck, Jack W. Lepisto, Alan Messner, and Paul F. Ritterman. Their participation is gratefully acknowledged.

CONTENTS

CONTENTS (Continued)

CONTENTS (Continued)

CONTENTS (Continued)

ILLUSTRATIONS

ILLUSTRATIONS (Continued)

ILLUSTRATIONS (Continued)

ILLUSTRATIONS (Continued)

TABLES

ACRONYMS

ADCCP	Advanced Data Communications Control
AMPS	Autonomously Managed Power System
BIA	Data Bus Interface Adapter
BIAH	BIA Handler
CPU	Central Processing Unit
CRC	Cycle Redundancy Code
CRT	Cathode Ray Tube (Video Display)
CSMA	Carrier Sense, Multiple Access
DCL	Desired Channel Loading
DMA	Direct Memory Access
DOD	Depth of Discharge
EOC	End of Conversion
EPS	Electrical Power Subsystem
EPSC	Electrical Power Subsystem Controller
FCS	Frame Check Sequence
HDLC	High-Level Data Link Control
I/O	Input/Output
ISO	International Standards Organization
LCC	Load Center Controller
LSI	Large Scale Integration
PL/M	Programming Language/Microcomputers
PMS	Power Management Subsystem
PSC	Power Source Controller
OC	Overcharge
OS	Operating System
RAM	Random Access Memory
ROM	Read Only Memory
RPN	Reverse Polish Notation
RTCH	Real-Time Clock Handles
SASU	Solar Array Segment Switching Unit
S/C	Spacecraft
SDLC	Synchronous Data Link Control
SOC	State of Charge
S/S	Subsystem
STC	Standard Test cycle
TDRSS	Tracking and Data Relay Satellite System
TQ	Timer Queue

1. INTRODUCTION

The growing size and complexity of spacecraft power systems coupled with limited space/ground communications necessitate increasingly automated onboard control systems. Efforts to address this problem have focussed on relatively low level algorithmic control, such as solar array orientation and shunt regulation, resulting in impressive improvements in both operations cost and system performance. Higher levels of control are effected through ground-based monitoring and commanding. Voltage and current limit selections are examples of this level of control. Further attempts at automating space power systems will need to address control functions at this higher level and beyond. Substantial potential exists to automate routine operations and maintenance functions and to aid in the diagnosis and recovery from system anomalies.

NASA and military applications for high power large space platforms and mannable space stations are presently being planned. Each satellite will provide an electrical power utility for a wide variety of missions and payloads over several decades of satellite life (30 years or more). As such, these satellites must provide operational flexibility and load adaptability in a cost-effective manner with the capability for evolutionary growth and performance upgrading as mission profiles change. These large, long-life spacecraft present new challenges from the standpoint of increased system complexity, new maintenance strategies utilizing space shuttle resupply, increased ground station operational burden, and survivability with unattended operation. Autonomous power management has been proposed as a solution to these new challenges.

2. EXECUTIVE SUMMARY

The concept of autonomous operation, as defined for this project, extends beyond the present day control strategies for safe haven and automated operation. Safe haven operation provides for the shutdown of equipment upon an out-of-tolerance performance condition. A ground command is required to reinitiate or override the shutdown condition. Automated operation takes the safe haven strategy one step further by switching to a redundant unit upon the out-of-tolerance operating condition. Equipment shutdown occurs (safe haven) if the redundant unit operates with an out-of-tolerance condition, and a ground command is used at that time to reinitiate operation.

Autonomous operation extends beyond this automated operation control strategy by continuing to operate in a degraded mode with the best of the available units. Maximum usage of the remaining available power is made by reducing or reallocating loads. A retrenchment to the safe haven mode occurs only upon extensive performance degradation causing loss of autonomous control. Ground commands are then used as backup or for unforeseen failure modes.

Implementation of autonomous operation can provide substantial benefits:

a) Power availability is increased.

b) Critical conditions are minimized by the enhanced speed of recovery after fault isolation.

c) The power subsystem is saf when out of communication with terrestrial control.

d) The level and cost of operational support is reduced.

e) The quality of power service is improved.

In addition, the equipment providing autonomous operation may be further utilized to:

a) Extend equipment life through automated load management strategies.

b) Fully allc..e the available electrical power generation capacity.

c) Manage the consumption rates of stored energy to enhance system performance, extend equipment or mission life, or maximize output power.

The value of these benefits typically outweigh the cost and weight of the additional onboard computational equipment and sensors required to implement autonomous control.

2.1 SCOPE

The scope of this task is to develop the concept for an autonomously managed power system (AMPS) based on the electrical power subsystem (EPS) reference design of Volume 1, to initiate the design and development of the essential components of the power management subsystem (PMS) portion of the autonomously managed power system, to initiate development of the associated algorithms necessary to perform the required power subsystem monitoring and control functions, and to identify the associated technology drivers.

The primary functions of autonomous operation are to recognize a fault, isolate the fault quickly, and reconfigure the power subsystem to recover functional operation so that the satellite mission may continue. Implementation of these functions requires adequate sectionalization of critical equipment, alternate operational paths or modes, and/or redundancy. A set of management strategies (algorithms), that are preplanned, are required to define the operating options and decisions within the inherent isolation and recovery limitations for the conceivable faults. Sensors (to acquire subsystem operational and state-of-health data), data and program storage, computational equipment, and subsystem control actuators (to implement reconfiguration decisions) are required to implement these functions. This study task addresses the configuration of PMS equipment to enhance automation options and operational strategies.

2.2 REQUIREMENTS

The contractual statement of work outlines a four-step program to define a power management subsystem (PMS) for development and demonstration of autonomous operation of multi-hundred-kilowatt electrical power systems for spacecraft:

a) Develop an autonomous power management concept compatible with the 250 kilowatt electrical power system definition.

b) Assess the technology status to define component and process deficiencies and the technology drivers for a 1985-86 technology readiness and a 1988 initial operational capability.

c) Identify the power management strategies control laws, and algorithms required for automated control including definition of monitors and sensors, control switches and effectors, and computational capacity and speed.

d) Initiate breadboard development hardware and software development to demonstrate the power management concepts, and ultimately integrate this PMS with a multi-channel, high-voltage, high-power test facility representative of a portion of a 250 kilowatt power system.

Management requirements and functions were subsequently defined to identify the operational roles of a PMS. These requirements and their related functions were grouped into a matrix (Table 2-1) according to potential geographic management centers (power sources, load centers, and integrated electrical subsystem) and by perceived operational categories (mission operation, maintenance, and degradation/fault accommodation). From this matrix, the management algorithm list and the topology and hierarchy of the distribution of power management control were developed.

2.3 ALGORITHM DEVELOPMENT

The functions that are performed within the EPS must be transformed into algorithms during the development of autonomous power subsystem management. Algorithms are the defined processes, rules, or procedures that are assigned to the functions to assure the realization of a desired output from given input conditions. Usually, algorithms are a sequence of formulas and/or algebraic logical steps that calculate a particular result or determine a given task.

The major algorithms identified for an autonomously managed power subsystem are listed in Table 2-2. These algorithms are derived directly from the functions that are performed in the EPS, as shown in Table 2-1. These algorithms are grouped into the three main categories of power source management, load center management, and EPS management.

Table 2-1. Management Requirements and Functions

Management Category	Operational Category					
	Mission Operation		Maintenance		Degradation/Fault Accommodation	
	Requirement	Function	Requirement	Function	Requirement	Function
Power Source Management	• Charge Battery	• Battery Charge Current Control	• Periodic Reconditioning • Trend Projections	• Battery State of Health Determination and Analyses	• Degradation Detection • Modify Operating Conditions • Remedial Reconditioning	• Solar Array Status • Battery State of Health Determination
Load Center Management	• Distribution of Power	• Command Processing • Switch and Load Bus Monitoring	• Replacement Scheduling	• Switch and Load Bus Monitoring • EPS State of Health Analyses	• Failure Detection	• Switch and Load Bus Monitoring
EPS Management	• Provide Energy Balance • Load Bus Assignments	• Energy Planning and Allocation • Load Bus Assignments	• Replacement Scheduling	• Power Subsystem State of Health Analyses	• Degradation Detection • Modify Operating Conditions • Controller Anomally --- • Solar Array Power Reallocation	• EPS State of Health Analyses --- • Energy Planning and Allocation

Table 2-2. List of Algorithms

Power Source Management	Load Center Management
• Battery Charge Control	• Command Processing
• Battery State of Health	- Circuit Breaker Programming
- Reconditioning	• Switch and Load Bus Monitoring
- Trend Projection	- Fault Definition
• Solar Array Status	
- SASU Switch Status	

EPS Management

- Energy Planning and Allocation
  - Solar Array Power Reallocation
- Load Bus Assignments
- Power Subsystem State of Health
  - Replacement Scheduling
  - Controller Anomally

In general, a first iteration of each algorithm was developed to the functional definition as described in Section 5. Portions of the load bus assignments, command processing, and switch and load bus monitoring algorithms were developed through the implementation and testing stages. These algorithms were developed to verify operation of the load center controller and EPS controller (EPSC) that were developed during this phase of the program. These algorithms were implemented in FORTH, a high-level programming language that was determined to be best suited for the AMPS application.

In addition to the functional algorithms listed in Table 2-2, each controller requires an executive algorithm. The executive algorithm manages the overall operation of the controller. Functions provided by the executive algorithm are process scheduling, management of common data buffers, timer services, and data bus adapter interfacing.

2.4 POWER MANAGEMENT SUBSYSTEM DESCRIPTION

The power management subsystem (PMS) monitors and controls the electrical power subsystem (EPS) from the power source to the loads so that the monitoring, processing and decision making, command, and memory functions can be efficiently and predictably performed. The PMS consists of power source controllers (PSCs), load center controllers, (LCCs), electrical power subsystem controller (EPSC), the data bus between the controllers, and the interfacing input/ output (I/O) circuitry between each controller and its respective sensors and effectors (typically switch gear).

A decentralized data processing approach was selected for the PMS based on the EPS algorithms that are to be performed and the control hierarchy trade studies and analyses. The PMS is a decentralized processing system from the standpoint that the PSCs, LCCs, and I/O circuitry are distributed throughout the spacecraft at various functional centers. The distributed processing concept for an autonomously managed EPS is shown in Figure 2-1. The power system controller and the redundant unit are located in the spacecraft control center where power subsystem information is displayed for onboard personnel and is also available for further processing to ground through telemetry and command subsystems. Power system commands are also received through these same channels and are routed to the functional centers via the data bus. Power source controllers are located at power generation centers where solar array power is integrated with energy storage device power to form the main power buses for the spacecraft. The main buses are then routed to the various spacecraft load centers where the LCCs are located. The LCCs provide power control commands for load bus and payload operations and monitor the implementing switch gear operational status. Distribution voltages and currents are monitored and power levels calculated.

Three types of microprocessor-based controllers are defined in the selected PMS concept: EPSC, PSC, and LCC. The primary function of each controller is to generate logical control decisions for the operation of the EPS and to manipulate sensor and other input data. To accomplish this, logical processing and data storage functions are required. The microprocessor and memory elements of each controller implement these functions. These elements also provide the mechanization to program the algorithms

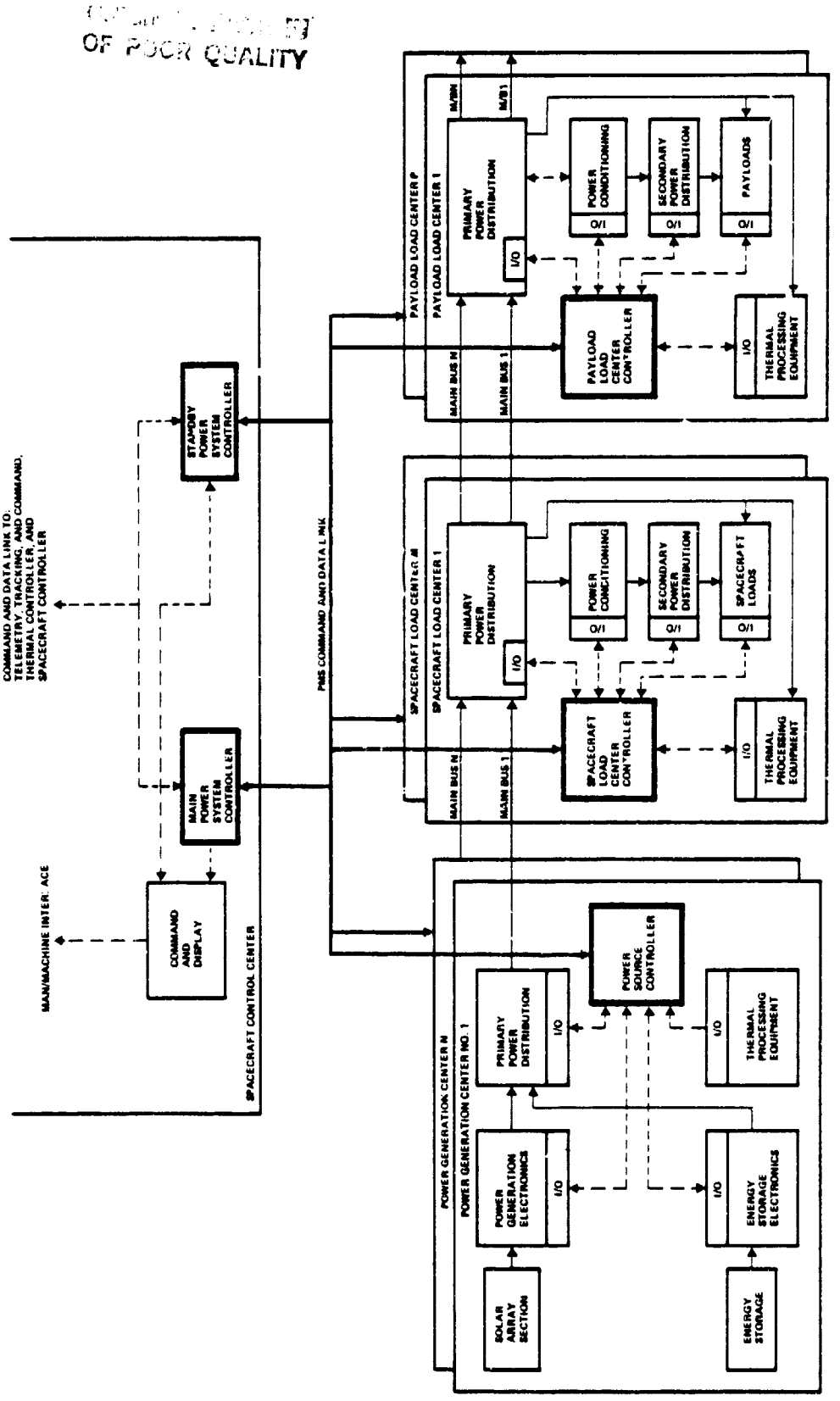


Figure 2-1. Distributed Processing for an Autonomously Managed Electrical Power System

that govern the control operations. The remaining major elements in each controller are I/O oriented and dependent upon the execution and implementation requirements of the control functions assigned to each controller.

Each controller may communicate with the other controllers over a common data bus. A bus interface adapter (BIA) in each controller implements this communication path by formatting the output data into an appropriate message structure (rate, format, time slot). The BIA also recognizes a proper input message structure and decodes this message into data/command format. The BIA consists of four sections: bus I/O circuits, bus protocol logic, data buffers, and direct memory access I/O logic.

Additional elements are included in each controller to provide peripheral but practical features such as power-on reset, timing checks to escape from a nondefinitive or excessively lengthy processing routine, and interrupt procedures to prioritize processing routines. A digital I/O port is provided on each controller for a serial data interface to provide for human interactions with the system. These peripheral devices, such as printers and video/keyboard terminals, may be connected to the system during development, testing, and demonstration.

The I/O circuitry consists of sensors, actuators, and their signal processing circuitry. Monitoring functions are implemented by the use of sensors. Sensors measure various system parameters and generally transform these signals into analogs that are usable by the electronic signal processing circuitry. The outputs from the sensors are analog signals that are proportional to the input parameters or status devices such as relay contacts that indicate change in status. Typically, analog sensors are voltage, current, pressure, and temperature transducers.

The signal processing circuitry gathers data from the sensors and processes this data so that it can be used by the PSC or LCCs. The signal processing circuitry is, therefore, performing a portion of the processing and decision function. A second responsibility of the signal processing circuitry is to process commands from the controllers and route them to the appropriate actuators. Specific functions of the signal processing circuitry are multiplexing of sensor channels, analog-to-digital conversion, digital-to-analog conversion, encoding and decoding, and logic functions.

The I/O circuitry also provides the interface with onboard personnel for the monitoring and control of the power system. Power system conditions are displayed by means of status indicators and information is provided through readout devices such as video displays. Commands are issued through input terminals such as keyboards.

2.5 HARDWARE DEVELOPMENT

An EPSC and a LCC were developed during this phase of the program. Each controller consists of a microprocessor, memory, I/O circuitry, and a BIA. The microprocessor that was selected for the controllers is a Texas Instruments 9900 unit with 64K bytes of memory capacity. Up to 20K of the memory can be read only memory (ROM) with the remaining portion random access memory (RAM). The microprocessor, memory, and I/O circuitry were assembled from commercially available Texas Instruments printed circuit cards. The BIA is being developed on a TRW internal research and development program. The cards are assembled as a unit in a Texas Instruments TM 990/530 sixteen-slot card chassis mounted in a commercial 19-inch rack. A picture of the overall system is shown in Figure 2-2.

2.6 STUDIES AND ANALYSES

A series of studies and analyses were performed to develop the details of an effective PMS for the 250-kilowatt power system:

a) Autonomy: the evaluation and selection of onboard control rather than terrestrial control.

b) Hierarchy: the development of a distributed approach to control processing rather than a central computer control.

c) Data Network: the evaluation and selection of a global bus architecture for communication among the PMS distributed controllers.

d) Operational Strategies: the evaluation of several operational options utilizing the PMS to reduce power system operational costs (i.e., equipment resupply).

These studies developed an effective PMS concept for the implementation of large complex power systems (250-kilowatts). In addition, major advantages and benefits from power management were identified and quantified. Life cycle costs of space power can be reduced, power availability

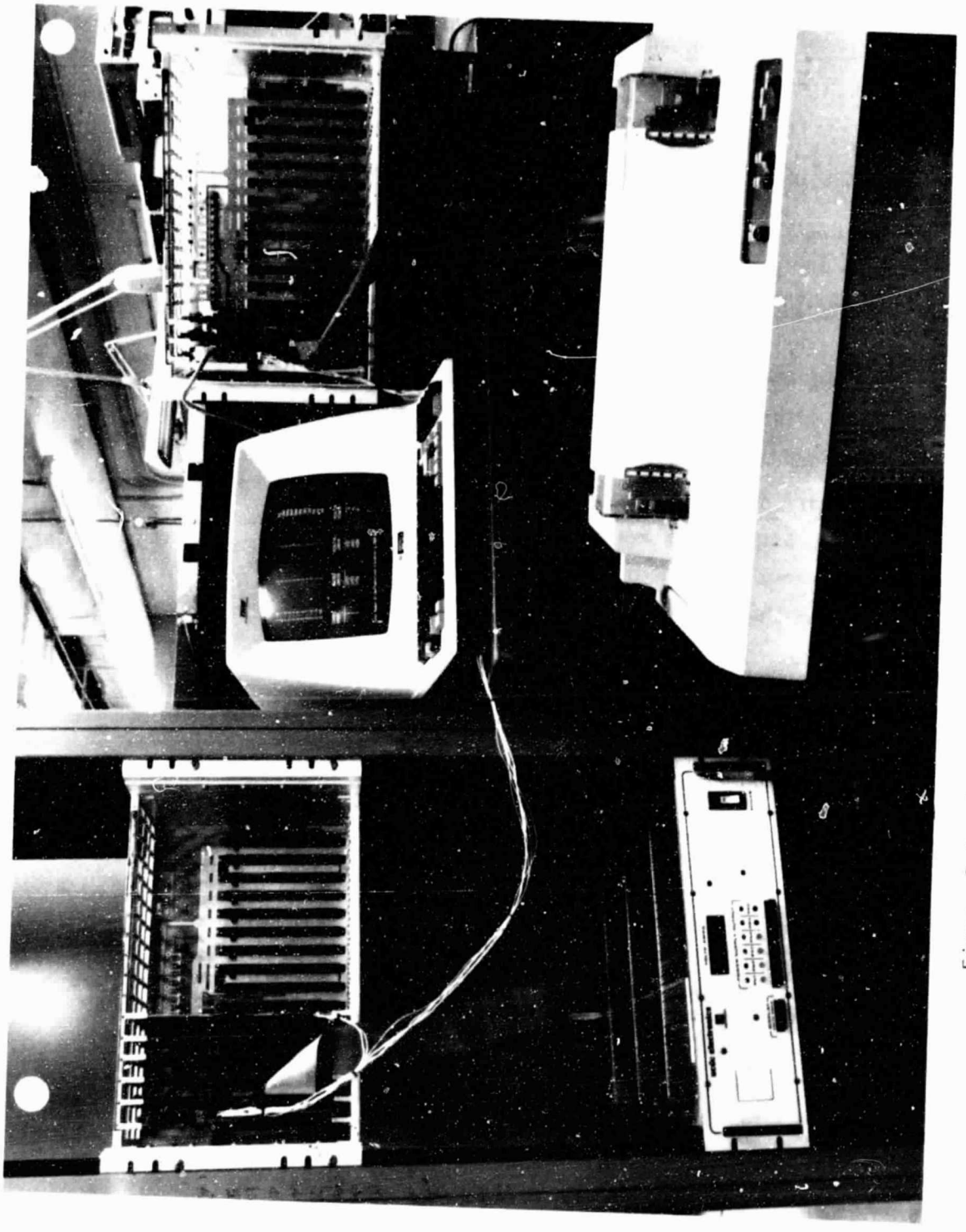


Figure 2-2. Power Management System Controllers

is enhanced as the safe haven operation is essentially avoided, and the spacecraft electrical power system becomes user friendly - a utility.

2.6.1 Autonomy

Automated control may be implemented by onboard in-flight computational capability (autonomous control) or by command and data telemetry between the spacecraft and a terrestrial control center with the computational capacity. Autonomy is required for rapid response to malfunctions or when the spacecraft or satellite communication time is protracted (by distance or orbit view angle). However, a safe haven mode may be implemented for these conditions. To more fully evaluate these options, a cost trade was performed to compare autonomous operational costs with terrestrial control costs (Figure 2-3). The projected costs included: (1) the additional onboard microprocessors for autonomous operation, (2) the expanded telemetry equipment for command and data transmission for terrestrial control, (3) the projected costs for one dedicated ground station for terrestrial control, and (4) the estimated annual labor costs. Based on these costs, the autonomous approach provides a significant economic benefit both initially and over a protracted time period.

2.6.2 Power Management Subsystem Hierarchy

Trade-off studies were performed to determine the control hierarchy for the PMS within the electrical power system. Centralized, distributed, and several hybrid concepts were considered. The three-tier hybrid concept (Figure 2-4) was selected as the recommended hierarchy. This selection requires modest computational power in each controller, provides a well-defined interface with the spacecraft and other subsystems, provides subsystem autonomy for assembly, testing, and algorithm development, minimizes the quantity and distance of sensor data transfer, incorporates subsystem control to simplify controller interactions and communications, and accommodates growth in both energy storage (number of channels) and load center quantity.

2.6.3 Data Network

A global bus architecture was selected as the data bus concept for the power management system because of its flexibility, reliability, failure tolerance, data rate accommodation, and equipment economy. This

Figure 2-3. PMS Reduces Monitoring and Control Costs

ADVANTAGES

- REASONABLE COMPUTER SPEEDS
  - EPSC 700 kHZ
  - PSC 200 kHZ
  - LCC 280 kHZ
- COMPUTATIONAL POWER GROWS WITH SYSTEM SIZE
- ALLOWS MODULAR BUILDUP AND EASIER EXPANSION
- SELF CONTAINED – ALLOWS FOR COMPLETE CHECKOUT AT SUBSYSTEM LEVEL
- MINIMIZES DATA TRANSFER OVER LONG DISTANCES (UP TO 300 FEET) THROUGH HIGH EMI (HOSTILE) ENVIRONMENT
- SIMPLIFIES INTERORGANIZATION COMMUNICATION

Figure 2-4. Selected PMS Configuration

architecture uses a single twisted pair electrical path (or single optical path) to minimize wiring between controllers. In this scheme, any single controller can communicate directly with any other controller on the data bus. Broadcast messages to all controllers can also be transmitted. The architecture works in conjunction with a distributed-control, time-sequential data bus contention resolution scheme and the International Standards Organization (ISO) high-level data link control (HDLC) data bus protocol to provide the overall data communications design. The global architecture is inherently flexible. Growth of the data bus system and/or modification of the system topology is easily accomplished by simple addition (or deletion) of the related BIA units and the associated data bus wire length without disturbing existing equipment on the bus and data bus routing.

2.6.4 Operational Strategies

Two operational strategies were investigated to reduce resupply costs: battery depth of discharge control and thermal fluid pump control. Potential cost savings are beyond $5M for balanced battery operation (Figure 2-5) and $5M for matching pump operation to the load profile (Figure 2-6).

2.7 CONCLUSIONS

The autonomous power management task of this study established the need for electrical power system management, developed a power management subsystem concept, identified the major algorithms for power system control, assessed the status of the supporting technology, and initiated a demonstration of the management concept through software development of representative algorithms and assembly of breadboard hardware. The power management subsystem controller was developed for the 250-kilowatt reference electrical power system for a large space platform in low earth orbit. This management concept incorporates flexibility for modular power growth, payload variety and growth, and spacecraft mission redirections. Consequently, the power management concept is adaptable to a wide variety of satellite applications and missions with a broad range of power levels, distribution voltage levels and forms, energy storage devices, and power source types.

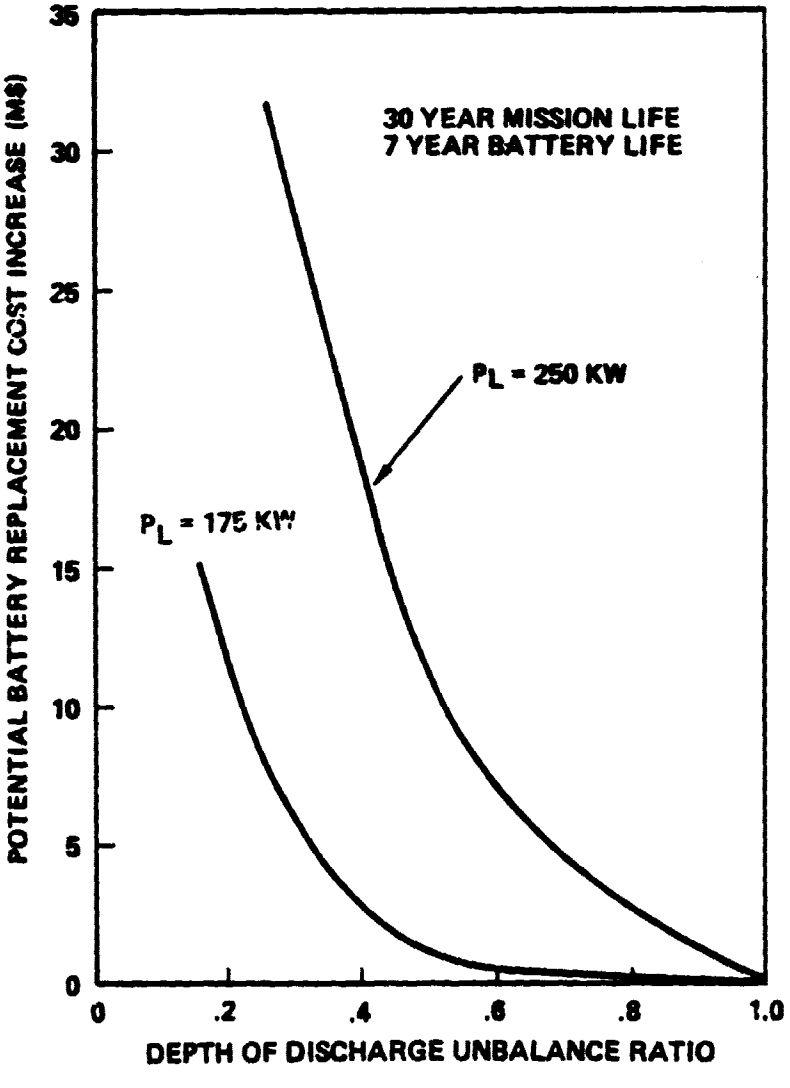


Figure 2-5. Battery Load Balancing Reduces Cost

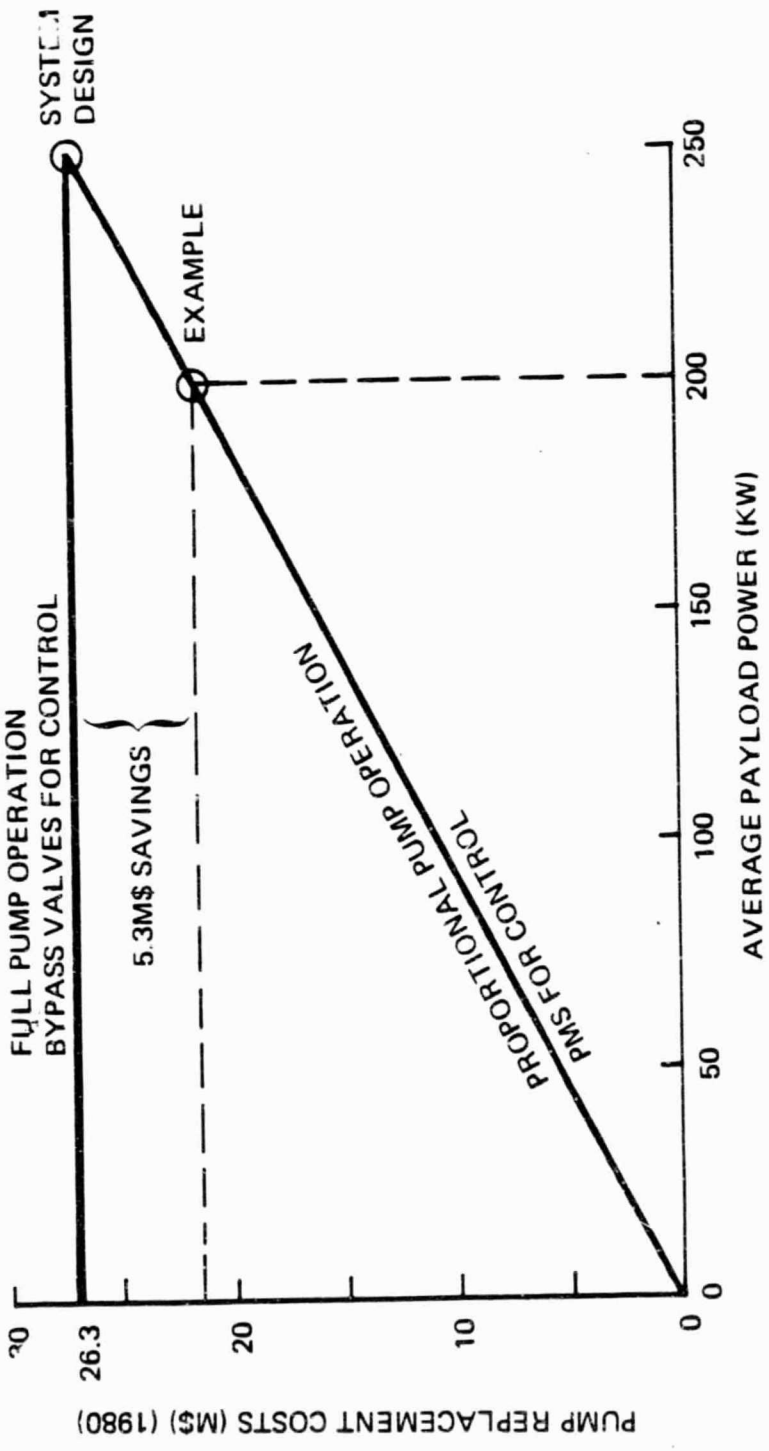


Figure 2-6. Autonomous Power Management Can Reduce Coolant Pump Costs

The power management concept utilizes a decentralized data processing approach to achieve autonomous operation of the electrical power system. The power management algorithms were identified, and software development was initiated for the load bus assignment, load switch command processing, and switch gear monitoring algorithms. Demonstration of the power management system was initiated with assembly of an electrical power subsystem controller breadboard and a load center controller breadboard.

Power management provides a means to integrate and control effectively a large number of smaller power sources and energy storage devices and distribution equipment into a cohesive multi-hundred-kilowatt electrical power utility. This method of electrical power management implementation (algorithms and controller hardware) provides an attractive approach to near-term readiness for large space generation and applications. Development of very large power generation and energy storage elements and the corresponding very high current switchgear that would be required to achieve a multi-hundred-kilowatt electrical power system is thereby avoided. Consequently, power management is an enabling technology to near-term multi-hundred-kilowatt availability of electrical power systems for spacecraft.

The hardware of the power management subsystem is modular in nature which provides great system flexibility. Incremental assembly and future expansion of a power system is supported with minimum development and expansion costs, and resupply costs are reduced. Furthermore, the modular approach allows application of this power management concept to a broad range of system power ratings (50 to 500 kilowatts).

The application of autonomous power management technology is cost effective and provides enhanced operation:

a) Ground station support costs are reduced, and communication traffic requirements are minimized.

b) Degradation and/or failure modes are quickly and efficiently accommodated so that power availability to the payload is enhanced (minimizes safe haven occurrences).

c) Power generation and energy storage are capable of full utilization to maximize payload support.

d) Equipment life may be enhanced by implementation of automated operational strategies.

e) Acquisition of trend data and analyses thereof provide degradation and resupply projections to optimize refurbishment scheduling and minimize replacement occurrences yet maximize the aggregate power availability to payloads.

2.8 RECOMMENDATIONS

Technology developments for the PMS fall into two categories: algorithm development and hardware development. Of these, algorithm development for the EPS (management strategies and operational control laws) will require the longest time, and thus immediate effort thereon is urgently needed. The implementation of these preplanned decisions and operational strategies is critical to enable development and operation of large capacity power systems by combining and controlling numerous smaller capacity power components into a utility-type EPS for multi-hundred-kilowatt power levels in space.

Hardware development of microprocessors is progressing rapidly due to data processing requirements in the commercial sector. Although only one radiation hardened microprocessor is available at this time, several CMOS processors are coming on the market that appear adequate for multi-hardware-kilowatt power management systems. Data bus protocols are being developed by the commercial sector (for distributed processing applications in banks, airlines, etc.) that appear to be more than adequate for application to the power management system.

A large quantity of sensors is required to monitor the operation of a multi-hundred-kilowatt EPS subsystem for autonomous control; because of the large quantity, a need exists to develop lightweight, low power, accurate sensors for current and pressure data. Present devices are typically heavy, consume modest power, and provide accuracy of only 2 to 3 percent of full scale. Such accuracy is marginal for some control functions (e.g., battery charge control).

3. SCOPE

The goal of automated power management is to provide a payload-friendly, utility-type electrical power system in support of future space platforms and space stations. The autonomous power management task of this study is therefore structured to:

a) Establish the need for electrical power system management by identifying the risks and benefits of power management

b) Develop a management concept for high-power satellites using the 250-kilowatt power system as a reference

c) Assess the status of technology to support electrical power management consistent with a 1985-86 technology readiness, and identify technology drivers

d) Identify the major management strategies (algorithms) to control a complex electrical power system

e) Demonstrate the power management concepts by software development of representative algorithms and through assembly of breadboard controller hardware.

Trade studies, analyses, and evaluations are performed to develop the concept of an autonomously managed power system based upon the 250-kilowatt electrical power system reference design. Subsequently, the design and development of the essential hardware components of the power management subsystem are initiated, and the development of representative algorithms is initiated.

4. REQUIREMENTS DEFINITION

This section describes the set of requirements that are pertinent to the development of the PMS. These requirements include the general system requirements for the EPS and functional requirements for the PMS. The functional requirements for the PMS include operational, maintenance, and degradation/fault accommodation requirements.

4.1 ELECTRICAL POWER SUBSYSTEM REQUIREMENTS

The EPS requirements are the general requirements that define the EPS operational constraints.

4.1.1 Purpose

The EPS is a part of a large space platform whose purpose is to provide utility type electrical power (250 kilowatts) to a variety of undefined payloads. A utility-type power system is defined as one that provides electrical power from more than one energy source to a set of undefined payloads.

4.1.2 Mission Capability

The spacecraft will operate in low earth orbit (90 minute orbit, 36 minute eclipse) and be capable of servicing a wide variety of missions. Specific missions will be determined by the nature of the payloads.

4.1.3 Modes of Operation

The spacecraft will have two main modes of operation:

a) Manned operation with shirt sleeve environment for indefinite periods of time with periodic shuttle refurbishment

b) Autonomous, unmanned, and unattended operation for an indefinite period of time.

4.1.4 Operating States

The spacecraft will operate in predictable states and will be controlled by autonomous on-board management, with optional on-board command or ground command overrides.

4.1.5 Reliability

Three main reliability requirements will be applied:

1) Safety redundancy: The power system will be reliable to the point that life support requirements are met.

2) Mission Redundancy: No single point failure will fail the mission. However, partial power outages will be allowed and accommodated by redundant power buses.

3) Life Redundancy: Proper component deratings and high reliability parts will be incorporated. Failed equipment will be replaced via space shuttle refurbishment. This policy will provide extended operational lifetime of 30 years and longer.

4.1.6 Environmental

The spacecraft will be designed to space shuttle launch requirements with normal spacecraft design temperatures.

4.1.7 Output Capability

The EPS will be capable of providing 250 kilowatts to an undefined number of loads. The power system will be flexible in nature and capable of expansion.

4.2 POWER MANAGEMENT SYSTEM FUNCTIONAL REQUIREMENTS

4.2.1 Configuration

The PMS will be configured as shown in Figure 4-1. The PMS will be capable of providing two levels of management. The top level of management provides the overall EPS management and interfaces with the spacecraft controller. The second level of management consists of power source management and load center management.

These management areas will perform the following specific functions:

a) Monitoring functions that measure the system parameters.

b) Processing and decision functions that perform the sorting, transformation, manipulation, and interpretation of data and that determine which monitoring, control, and recording functions are to be executed.

c) Control functions that execute the desired changes to the network configuration.

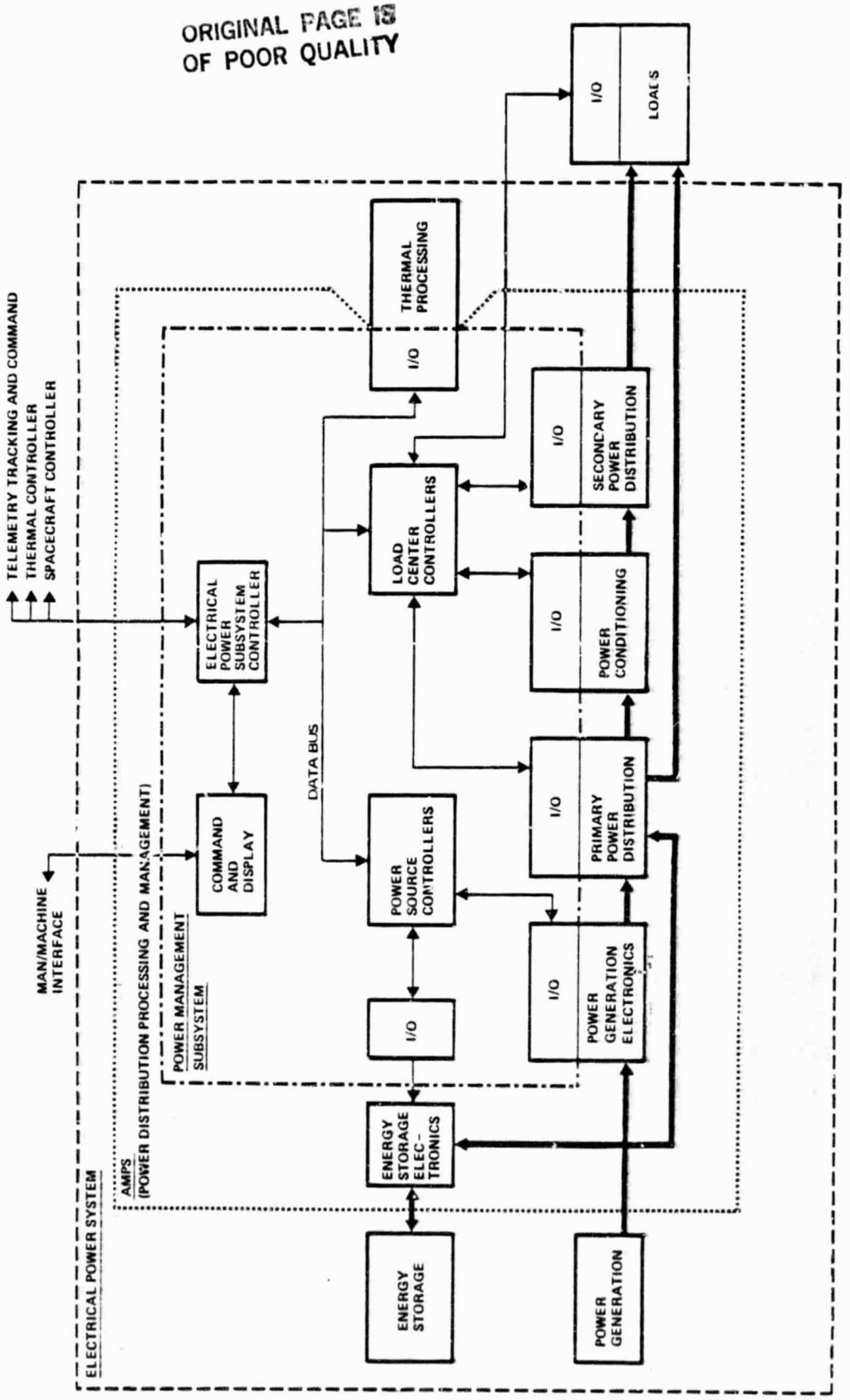


Figure 4-1. Power Management Subsystem Configuration

d) Recording functions that preserve the data needed for operational, trend projection, and fault analysis.

4.2.2 Scope of Responsibility

The PMS will provide the management of the complete EPS from power generation to loads by:

a) Decoding and executing commands from the central spacecraft controller

b) Acquiring power system data for on-board analysis and partial ground telemetry

c) Providing timing for synchronization of power system events

d) Providing high level processing for computational purposes, program and data storage, and execution of management of power sources and load centers.

4.2.3 Power Source Management Functional Requirements

Power source management provides the necessary monitoring, processing, decision, control, and recording functions for the power generation, energy storage, and switching devices and their associated power electronics hardware. The requirements that determine the implementation of the power management functions fall into the three main categories of operations, maintenance, and degradation/fault accommodation requirements.

4.2.3.1 Power Source Operational Requirements

The major operational requirement for power source management is to manage power generation and battery operation to support the loads and to recharge the power subsystem batteries. The power management system will provide a method to charge each nickel-hydrogen battery individually in the power subsystem based on a low-earth orbit mission. The charging algorithm will operate in conjunction with the energy planning algorithm at the EPS management level to provide a charging cycle that may be orbital or multi-orbital. The algorithm will provide a means to determine the state of charge of each battery at all times. Battery current, voltage, pressure, and temperature will be monitored. Battery charge current will be controlled based on the state of charge of the battery by varying the solar array output current (i.e., the number of solar cell string segments

connected to each battery bus). The battery parameters, including the battery state of charge, will be stored.

4.2.3.2 Power Source Maintenance Requirements

The major maintenance requirement for power source management is to maintain the state of health of each battery by analyzing battery performance during normal operation, performing periodic reconditioning, and making trend projections to determine when the batteries must be replaced. The reconditioning algorithm will work in conjunction with the energy planning algorithm to determine when reconditioning can be performed during the mission. The algorithm will perform both a charge and discharge reconditioning routine. Battery cell analysis will be performed during normal operation as well as the reconditioning routines. The cell analysis will be based on a statistical distribution of cell voltages. Battery current, temperature, cell voltage, and pressure will be monitored. Battery charge current will be monitored according to the cell analysis results. Cell anomaly data will be stored along with the associated battery current and temperature.

Battery trend projections will predict degradation and determine battery replacement needs based on an analysis of battery age, capacity, and cell data. An alert will be sent to the ground when a required battery replacement is approaching. Sufficient trend data will be stored for answering ground interrogations.

4.2.3.3 Power Source Degradation/Fault Accommodation Requirements

The PMS will determine degradation of the solar array, solar array switching unit, and batteries. Solar array degradation will be determined by measuring its available output power at a predetermined battery voltage. Degradation of the solar array switching unit will be determined by accounting for any solar array switches that have failed. Battery degradation will be determined from cell analysis data. Remedial battery reconditioning will be scheduled based upon degradation data. Battery capacity will be determined and sent to the energy planning algorithm at the EPS management level to accommodate load management. All status information will be stored within the PMS.

4.2.4 Load Center Management Functional Requirements

Load center management provides the necessary sensor monitoring, data processing, and decision, control, and data storage functions for the power distribution circuitry and the power conditioning circuitry (if required) located in the spacecraft load centers. The requirements that determine the implementation of the power management functions fall into the three main categories of operational, maintenance, and degradation fault accommodation requirements.

4.2.4.1 Load Center Operational Requirements

The major operational requirement for load center management is to distribute power to each of the load buses. The PMS will provide commands to programmable resettable circuit breakers that will allow for connection of the load bus to selected power channels. The status of each circuit breaker will be monitored and stored. Also, commands will be provided to set circuit breaker trip points. Switch currents, channel currents, and load bus voltages will also be monitored.

4.2.4.2 Load Center Maintenance Requirements

The major maintenance requirement for load center management is to monitor circuit breakers and sensors within the load centers for failures. The status of failed circuit breakers and sensors will be gathered in the load center and this information will be transmitted to the EPS controller via the data bus.

4.2.4.3 Load Center Degradation/Fault Accommodation Requirements

The major degradation/fault accommodation requirement for load center management is to detect failures of the circuit breakers and sensors within the load centers. The power management system will be capable of detecting the status conditions listed in Table 4-1. Fault information will be transmitted to the EPS controller for EPS state-of-health monitoring.

4.2.5 Electrical Power Subsystem (EPS) Management Functional Requirements

EPS management provides the necessary monitoring, processing, decision, control, and recording functions of the overall power subsystem. The requirements that determine the implementation of the power management

Table 4-1. Load Center Status Conditions

Status Type	Status Condition
0	No fault, load on
1	No fault, load off
2	Breaker open
3	Breaker closed
4	Breaker status open
5	Breaker status closed
6	Current transducer, no output
7	Current transducer, full output
8	Load undercurrent, $I_L < I_L$ min
9	Load overcurrent, $I_L > I_L$ max
10	Voltage transducer, no output
11	Voltage transducer, full output
12	Bus undervoltage, $V_L < V_L$ min
13	Bus overvoltage, $V_L > V_L$ max
14	Last command off
15	Last command on

functions fall into three main categories of operational, maintenance, and degradation fault accommodation requirements.

4.2.5.1 EPS Operational Requirements

The major operational requirements for EPS management are to provide energy balance for the power subsystem and to assign the load bus locations to the power channels. The power management system will provide an adequate energy balance for each channel and for the overall system. The energy balance algorithm will be operative during both sunlight and eclipse modes of operation to provide proper battery charging. Both orbital and multiorbital energy balance routines will be provided. The power management system will provide the capability to deactivate nonessential loads on a priority basis and to notify the spacecraft computer if proper energy balance cannot be attained. The energy balance algorithm will determine the desired loading on each power channel.

The load assignments function will determine the distribution system configuration which minimizes the mismatch between the desirable channel loading and that which is physically realizable. The power management system will be capable of making the load assignments based on system priorities and constraints such as the switchability of loads and the noise sensitivity of the loads. The load assignments function will be capable of determining that the selected configuration has been implemented. If the selected configuration has not been implemented, the load assignments function will be capable of reassigning the loads based on the new system constraints.

4.2.5.2 EPS Maintenance Requirements

The power management system will determine the overall state of health of the power subsystem. The power management system will identify when wearout of power subsystem components is imminent based on the subsystem state of health and will alert the ground station of these projections for subsequent replacement schedules. The power management system will store sufficient trend data to answer ground interrogation of the projected replacement requirements.

4.2.5.3 EPS Degradation/Fault Accommodation Requirements

The power management system will determine the overall state of health of the power subsystem. Information will be received from the power source and load center controllers based on degradation/fault accommodation requirements for these management areas. The EPS controller will maintain status tables for all electrical power subsystem operating conditions. The power management system will be capable of modifying system conditions such as performing emergency load shedding in order to maintain the electrical power subsystem in a proper state of health. The power management system will be capable of reallocating solar array power between power channels in order to maintain the electrical power subsystem in a proper state of health.

The power management system will be capable of determining failures within the power source, load center, and EPS controllers and of maintaining the system data bus in a proper operating condition.

4.2.6 Summary of Management Requirements and Functions

A summary of the requirements in each of the major management areas is presented in Table 4-2. The major functions that must be performed to meet each of the management requirements are also listed.

Table 4-2. Management Requirements and Functions

Management Category	Operational Category					
	Mission Operation		Maintenance		Degradation/Fault Accommodation	
	Requirement	Function	Requirement	Function	Requirement	Function
Power Source Management	• Charge Battery	• Battery Charge Current Control	• Periodic Reconditioning • Trend Projections	• Battery State of Health Determination and Analyses	• Degradation Detection • Modify Operating Conditions • Remedial Reconditioning	• Solar Array Status • Battery State of Health Determination
Load Center Management	• Distribution of Power	• Command Processing • Switch and Load Bus Monitoring	• Replacement Scheduling	• Switch and Load Bus Monitoring • EPS State of Health Analyses	• Failure Detection	• Switch and Load Bus Monitoring
EPS Management	• Provide Energy Balance • Load Bus Assignments	• Energy Planning and Allocation • Load Bus Assignments	• Replacement Scheduling	• Power Subsystem State of Health Analyses	• Degradation Detection • Modify Operating Conditions • Controller Anomally --- • Solar Array Power Reallocation	• EPS State of Health Analyses --- • Energy Planning and Allocation

5. ALGORITHM DEVELOPMENT

In the development of an autonomous power management subsystem, the functions that are to be performed in the electrical power subsystem must be transformed into algorithms. Algorithms are the defined processes, rules, or procedures that are assigned to the functions to assure the realization of a desired output from a given input. Usually, algorithms are a sequence of formulas and/or algebraic logical steps that calculate or determine a given task.

5.1 ALGORITHM DEVELOPMENT APPROACH

The general approach to developing algorithms is shown in Figure 5-1. The first activity is to define the system level requirements. These requirements include system level/operational requirements, electrical power subsystem functional requirements, performance requirements that are initially known, and system-imposed constraints. The requirements are defined by gathering inputs from various disciplines, such as operations, integration and test, environmental, components, subsystems, and the customer.

The next step is to define the top-level functions that must be performed in order for the system to meet the defined requirements, and to group the functions logically according to their interactions. Interfaces between these functions are defined as well as information or data flow between functions. The functional areas are fully described along with a first level of functional decomposition.

The next step is the detailed design activity which involves the examination of the functions and information structures defined previously, the expansion of the general functions expressed therein to provide more detail, and the grouping of this detailed information into subfunctions and their interfaces. This process proceeds through several iterations to the point where functional decomposition becomes procedural decomposition, i.e., where functions are described in terms of algorithms which affect them, and information structures are described in terms of data structures

Figure 5-1. Algorithm Development Approach

used by the algorithms. This iterative design process incorporates several well-known techniques:

a) Stepwise refinement. The sequence of steps which repeatedly breaks up tasks into subtasks and similarly refines data structures.

b) Top-down design. Beginning with a firm, fixed requirements specification, top-down design organizes and develops the control structure of a program through stepwise refinement.

c) Hierarchial decomposition. Provides guidelines on the amount of additional detail between successive refinement steps.

d) Modularity. Provides guidelines on the passing of information among functions on a given level (Reference 5-1).

Implementation and testing follow the detailed design. Implementation involves writing the algorithms in a computer programming language or code. Testing is performed to ensure that the system works according to the design requirements. Testing of software is usually performed bottoms-up starting with individual modules. The modules are gradually integrated together until the entire system is tested. Testing removes both programming language and logic errors that were introduced during the implementation stage. Testing usually involves running the program through all possible combinations of input and output constraints.

The last step is maintenance which is performed after delivery of the system to the user. Maintenance consists of on-site debugging to correct errors that were not detected during the testing stage and software upgrades to add new functions to the system.

The overall algorithm development approach is an iterative process in that each succeeding step may supply information and place additional constraints on the previous step. For example, it may be determined during the detailed design phase that certain functions should be done in hardware rather than software. This change could affect the function definition as well as the requirements definition step.

5.2 AUTONOMOUSLY MANAGED POWER SYSTEM (AMPS) ALGORITHMS

The major algorithms to be implemented in the autonomously managed power system are listed in Table 5-1. These algorithms were derived directly from the functions that are to be performed in the electrical

Table 5-1. List of Algorithms

Electrical Power Subsystem Management

- Energy Planning and Allocation
  - Solar Array Power Reallocation
- Load Bus Assignments
- Power Subsystem State-of-health
  - Replacement Scheduling
  - Controller Anomaly

Power Source Management

- Battery Charge Control
- Battery State-of-Health
  - Reconditioning
  - Trend Projection
- Solar Array Status
  - SASU Switch Status

Load Center Management

- Command Processing
  - Circuit Breaker Programming
- Switch and Load Bus Monitoring
  - Fault Definition

power subsystem, as shown in Table 5-1. These algorithms are grouped into the three main categories of power source management, load center management, and electrical power subsystem (EPS) management.

In general, a first iteration of each algorithm was developed to the functional definition, a step described in Section 5.1. Portions of the load bus assignments, command processing, and switch and load bus monitoring algorithms were developed through the implementation and testing stages. These algorithms were developed in order to verify operation of the load center controller and electrical power subsystem controller that were developed during this phase of the program as described in Section 7. These algorithms were implemented in FORTH, a high level programming language that was determined to be best suited for the AMPS application (see Section 8.7). Each of the algorithms is described in the following sections.

In addition to the functional algorithms listed in Figure 5-1, each controller requires an executive algorithm. The executive algorithm manages the overall operation of the controller. Functions provided by the executive are process scheduling, managment of common data buffers, timer services and data bus interface adapter (BIA) interfacing. The executive algorithm is described further in the following sections.

5.3 POWER SOURCE MANAGEMENT ALGORITHMS

The power source management algorithms provide the processes and procedures for the functions required to operate the power generation and energy storage devices and their associated power electronics hardware. The major algorithms are battery charge control, battery state of health, and solar array status. These algorithms are described in the following sections.

5.3.1 Battery Charge Control Algorithm

The 250-kilowatt power system includes nickel-hydrogen batteries for energy storage. A three-level charging approach is employed to recharge these batteries:

a) A high-rate charge essentially utilizing the net available power from the solar array (gross power output less loads)

b) A reduced charging rate suitable to essentially complete the battery recharge but at a safer rate to approach overcharge (an intermediate charge rate)

c) A trickle charge rate acceptable for overcharge during an entire sunlight period.

A battery charge control algorithm is employed to define and control the transfer to the next appropriate charging rate. Two approaches to battery charge control are provided for redundancy: (1) a state-of-charge control, and (2) a recharge fraction control.

The primary control method (Figure 5-2) is based upon state of charge as the basis for changing charging rates (high rate, intermediate rate, trickle rate). Meaningful state-of-charge calculations require knowledge of the instantaneous efficiency. Accordingly, efficiency is determined, in real time, by comparing the measured cell pressure data with the theoretical pressure calculated assuming 100 percent efficiency. Hence, cell pressure data is required for this approach.

Four major processing steps are used to assemble the charge control algorithm (Figure 5-2):

1) "Current sense" to determine the battery operating mode, recharge (+) or discharge (-)

2) "Increment Ah, SOC" to update the battery charge status

3) "SOC $\geq$ SOC 1" or similar test to trigger transfer to another charge rate

4) "Switch to trickle charge rate" or similar step to implement the transfer to the specified charge mode.

These building block steps are arranged in logical order (Figure 5-2) producing a series of four sub loops defining high-rate charge, intermediate charge, trickle charge, and discharge. In addition, a check of the level of overcharge attained is made, and the intermediate charge parameters and limits are adjusted accordingly. Also, initial parameters are defined for a fully charged battery. This precludes long term rundown of a battery due to inherent imperfections in the cumulation of incremental state-of-charge calculations.

This algorithm is continuously operational, but interrupts occur at each "increment" block during the time delay period. This allows

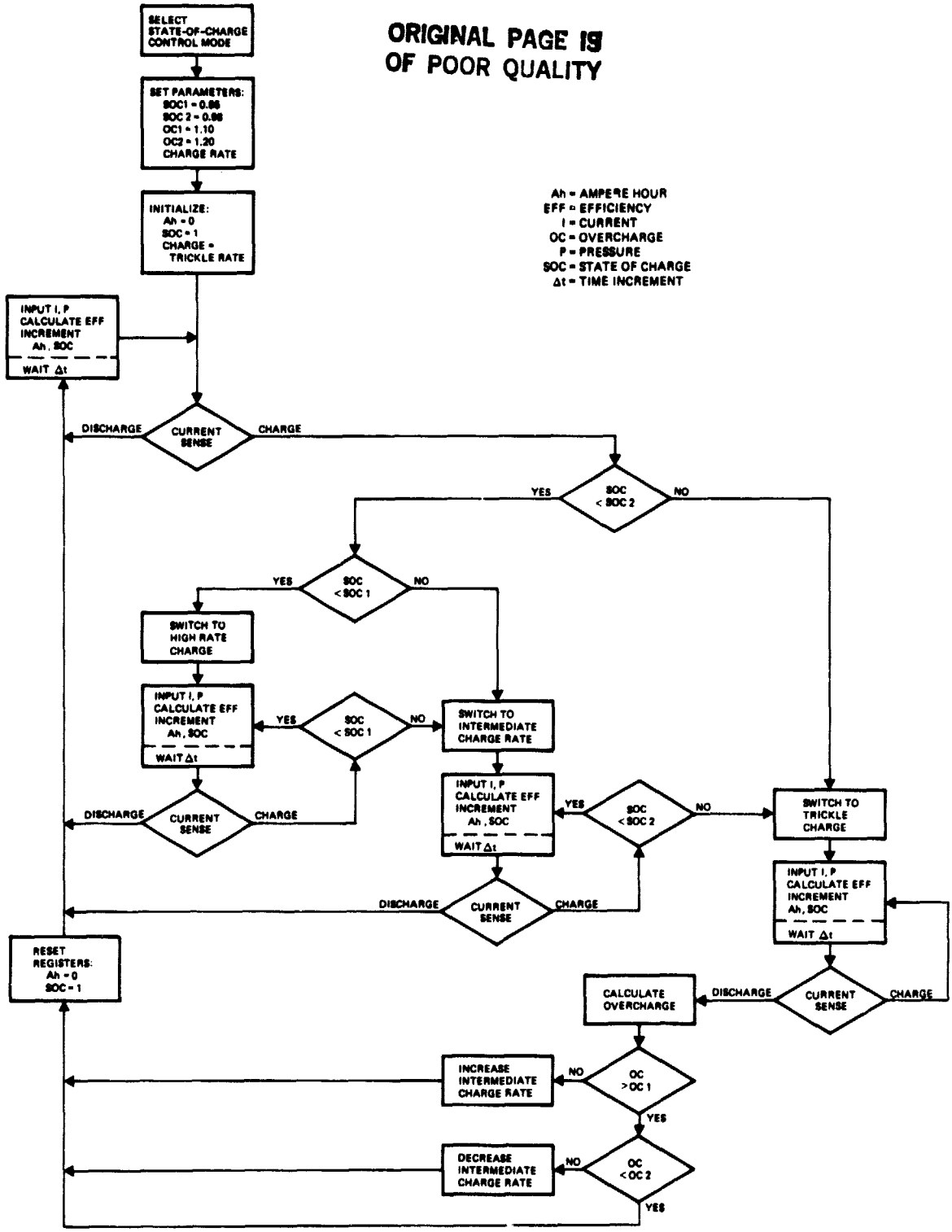


Figure 5-2. Battery Charge Control Algorithm Flow Diagram (State-of-Charge Method)

micro-processor time for other operations. After the time delay, the program resumes at the same point in the charge control algorithm. Executive control within each power source controller coordinates these interrupt operations.

The alternate charge control method is based upon ampere-hour integration without compensation for inefficiency of recharging. This control method (Figure 5-3) is structured similarly to the state-of-charge control method. However, this method utilizes ampere-hour calculations, accumulation, and comparisons for control decisions. State-of-charge calculations and the necessary efficiency determination from measured cell pressures are avoided. Hence, this method requires only battery current measurements.

These algorithms address control of normal batteries. Should abnormal cell operation develop (as determined from cell voltage and pressure measurements), charge control parameters (not the algorithm structure) are modified appropriately.

5.3.2 Battery State-of-Health Algorithm

The battery state-of-health algorithm has the following objectives:

a) To identify anomalous cell and/or battery performance

b) To save the data describing the anomalous performance

c) To identify anomalous situations requiring remedial action

d) To provide a historical data file for subsequent diagnostic evaluation.

The nature of a utility-type electrical power system precludes the assumption that the power profiles during any two discharge/recharge cycles are alike. This also precludes any simple trend analysis as the discharge rates and depth of discharge reached can vary significantly from cycle to cycle. A statistical approach to trend analysis with this data base requires complex modeling of each conceivable failure mechanism, its effect on performance parameters, and combinations thereof. However, simplified techniques to perform trend evaluation are included in the battery state-of-health algorithm.

Two approaches to identifying anomalous performance are utilized in the algorithm (Figure 5-4). In the first, a dispersion test, the measured

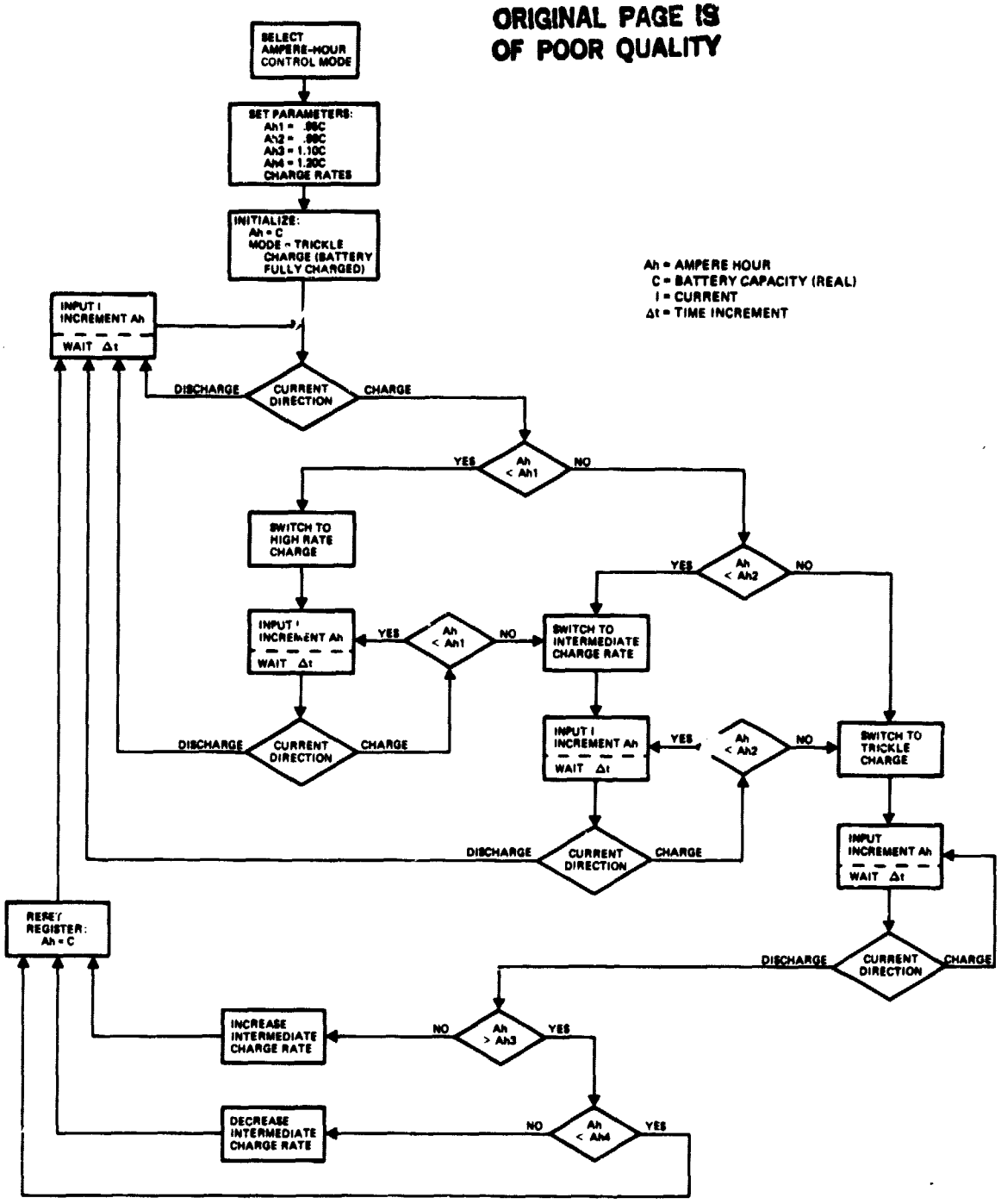


Figure 5-3. Battery Charge Control Algorithm Flow Diagram (Alternate Ampere-Hour Method)

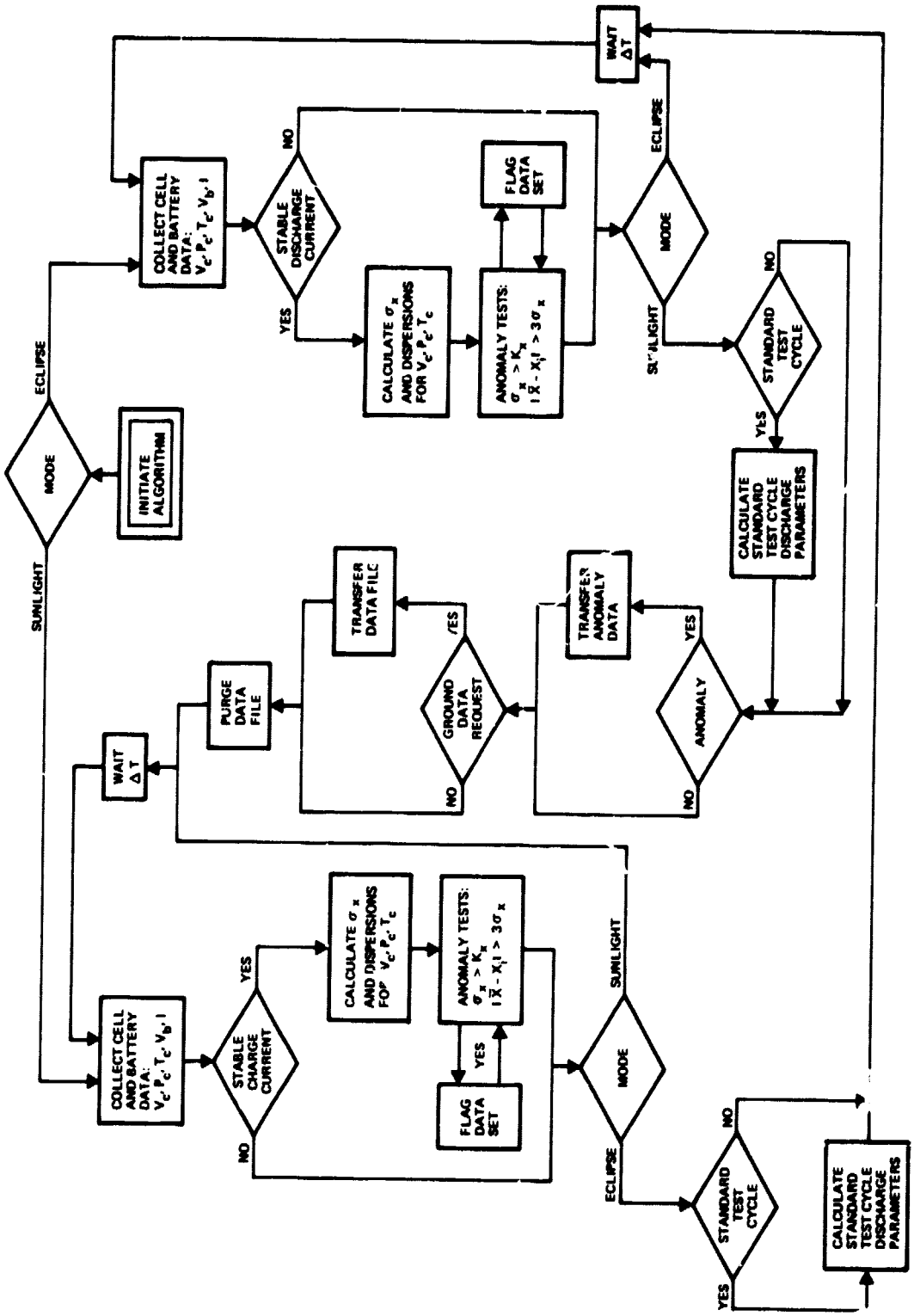


Figure 5-4. Battery State-of-Health Algorithm (Anomaly Detection)

attributes of voltage, current, pressure, and temperature for the 160 cells are taken over a relatively short time span, within several seconds. This set of 160 cell voltages, temperatures, and pressures are assumed to be distributed statistically normal provided the current was stable during the measurement interval. Therefore, individual values more than three* standard deviations from the mean represent anomalous cells. This dispersion approach, repeated approximately every minute, has the advantage of independence of the discharge/recharge rates and depth of discharge imposed by varying power profiles. The approach is shown graphically in Figure 5-5 for cell voltage analysis. Comparison of anomalous performance from one discharge/ recharge cycle to others having differing power profiles is invalid for trend analyses, however. Consistent discharge/recharge rates and depth of discharge (same power profile) from cycle to cycle are required for valid trend analyses.

The second approach postulates providing periodically a standard test cycle of essentially constant load current to a specified depth of discharge. This standard test load is provided by selectively allocating payload buses of essentially constant profile to the battery on test. Assuming one battery (of the 17 batteries in the 250-kilowatt system complement) is so loaded each eclipse cycle, each battery is evaluated by the standard test cycle once every 25.5 hours. The performance data from this standard test cycle are compared with the data of prior standard test cycles. Trends of cardinal data points such as mid-point and end-point values of the performance parameters are calculated and projected. Trends, exceeding predetermined acceptance ranges, are noted as anomalous. Data on these anomalous trends are stored for transfer to a ground station file for historical records and any further in-depth analyses deemed appropriate.

5.3.2.1 Files

Cell data and battery parameters are stored in three files: (1) a temporary on-board file, (2) an on-board rollover file, and (3) ground

*The actual limiting value for the acceptable spread from the mean requires determination from test data on nickel-hydrogen cells. (A large value for the standard deviation for voltage or pressure suggests general cell unbalance and reconditioning is required.)

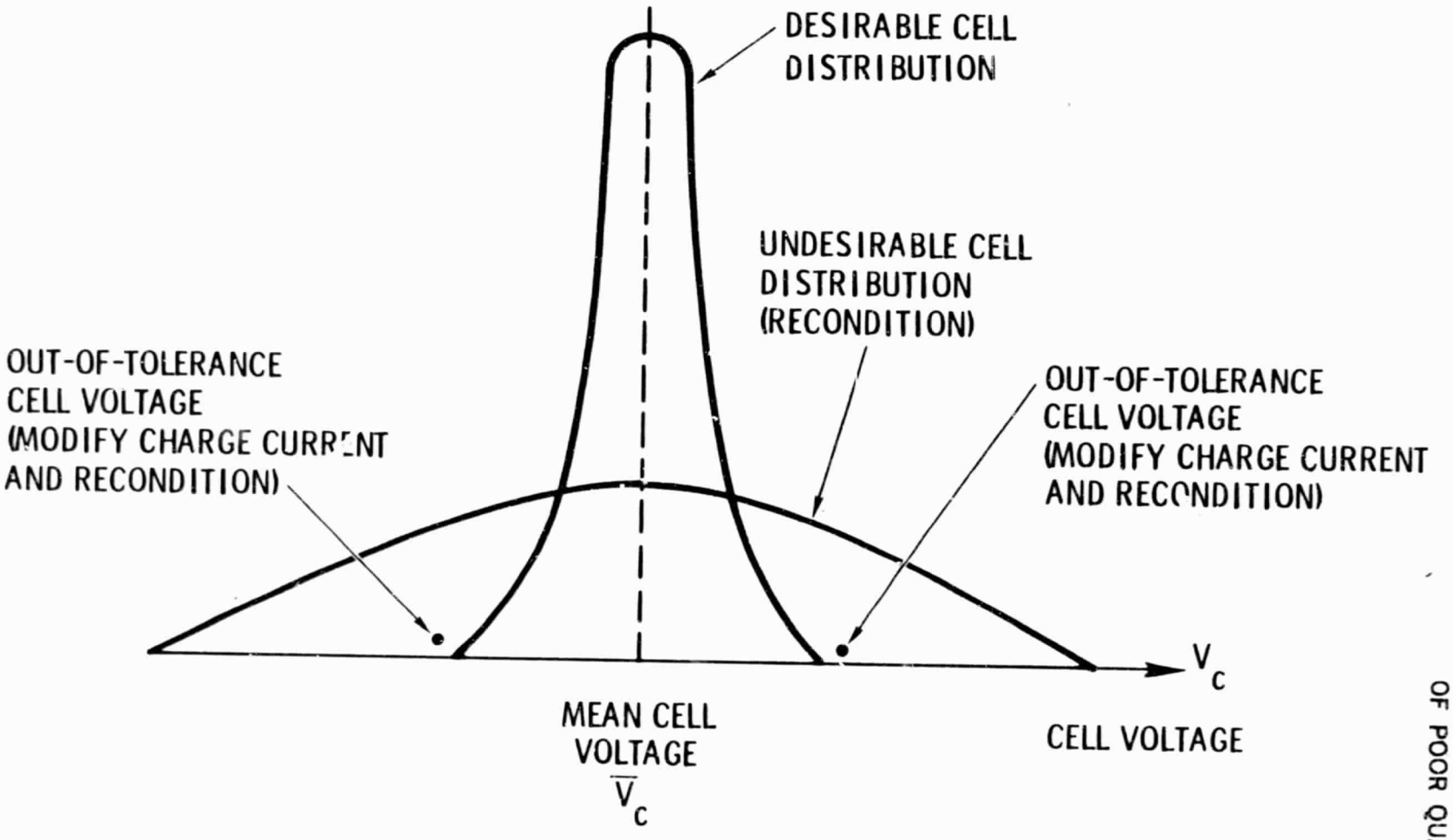


Figure 5-5. Cell Voltage Analysis Determines Battery State-of-Health

station file. A fourth file, containing anomaly data, is utilized for on-board trend analysis to select the proper corrective action response.

The temporary on-board file stores voltage, pressure, and temperature data for each cell, and voltage and current data for the battery for one recharge cycle and the subsequent discharge cycle. This results in 43,470 data points assuming one data scan per minute: 90 x (160 cells x 3 data parameters plus two battery parameters plus a time mark) = 43,470. This data is used for the dispersion tests to identify anomalous performance (approach one). Upon completion of a discharge cycle, the data is purged unless there is an anomaly. In that event, the data is transferred to telemetry storage for transmission to the ground control station for anomaly history records, and the data of the anomalous cell is stored on-board in the anomaly data file.

The on-board rollover file retains the pertinent parameters from the standard test cycles for trend analyses and predictions (approximately one cycle per day). This data includes mid-point and end-point values for the cell parameters of voltage, pressure, and temperature and for the battery parameters of current, voltage, and depth of discharge. This results in 13,552 data points assuming seven sets of standard test charge/discharge cycles are recorded for statistically significant trend analyses and projections:

(160 cells x 3 data parameters + 3 battery parameters + time mark)
x 2 for mid-point/end-point data sets
x 2 for charge/discharge data sets
x 7 cycles

= 13,552 data points

As new data is subsequently gathered and added to the file, the oldest data set in the file is discarded or transferred to telemetry storage for transmission to the ground control station for a historical record of trend data.

The rollover file gathers, in addition to the standard test cycle parameters, battery end-of-charge voltage and temperature (a mean value) and the depth of discharge (ampere-hours removed) for each eclipse cycle. This data requires only 3 bytes plus an identifying word (2 bytes) per

eclipse cycle. This requires a 640-byte field in the rollover file to accommodate 8 days of 16 eclipse cycles/day. As subsequent new data is gathered, the older data of this file block is transferred to telemetry storage for transmission to the ground control station (or discarded). These parameters provide a modest, but significant, history of normal performance cycles with little impact on file space requirements.

The ground station file is intended to produce a historical record as an expanding data base of nickel-hydrogen experience. The file receives three data types:

1) The periodic standard test cycle parameters from the on-board rollover file.

2) The charge/discharge parameters of those cycles having anomalous cell dispersions.

3) The end-of-charge battery voltage and temperature and depth of discharge for each operational battery cycle.

In addition, the ground control station may request the on-board temporary file data at the end of each charge/discharge cycle even when performance is normal. Such a request provides a complete and nearly continuous read-out of battery and cell performance.

Data for 30 days of operation aggregates approximately 60,000 bytes per battery, approximately 1,020,000 bytes per spacecraft with 17 batteries (250-kilowatt baseline). This is reasonable for terrestrial storage techniques (disk storage). The battery data must be accumulated and stored in an orderly and readily retrievable mode over the life of each battery. This data aids the battery engineers in evaluating battery problems and defining the appropriate corrective actions. This data is not intended for continuous status monitoring and plotting by ground personnel.

5.3.2.2 Data Compression

The battery and cell data required to diagnose performance correctly is voluminous. Several techniques are available to reduce the file space required for this data storage:

a) Use 8 bits (byte) instead of 16 bits (word) for parameter value recording.

b) Use statistical compression of the data points, i.e., calculate and save the mean and standard deviation rather than save every data point.

c) Apply a bit compression algorithm to reduce the bit quantity in the data from many 8-bit groups (the parameter bytes) to fewer nongrouped bits. Compressions of 3:1 to 20:1 have been realized with this technique.

Battery and cell parameters are defined as 8-bit values herein. The 8-bit value provides a digital resolution of 1 count in 256 counts (0.4 percent). Transducer accuracy is typically only 1 to 3 percent. Hence 1-byte data parameters are adequate.

Cell and battery data parameters aggregate approximately 58,000 bytes of on-board memory capacity. Typical processor memory space is 64K bytes without expansion provisions or bulk storage addition. The remaining 6K bytes is projected to be marginally adequate for battery control programs considering that the battery control software used on another study is implemented with approximately 3K bytes of program memory and can be purged to approximately 2K bytes. Additional program file space of 14K bytes is potentially available by reducing the temperature data from one recorded measurement per cell (160 per battery) to four parameters per battery (mean value, standard deviation, highest temperature, and lowest temperature). This provides a generous 20K-byte program file. Selection of any specific data compression or bit compression approach is left to subsequent studies. Additional bulk data storage may be as cost effective as these compression techniques.

5.3.3 Solar Array Status

Each of the 17 channels of the power subsystem contains a portion of a solar array switching unit that switches solar array sections to regulate the solar array outputs with either a current or voltage as the controlling parameter. The solar array panels are subdivided into many sections interconnected by solid-state switches. The solar array status algorithm monitors the solar array sections for failures, determines output capabilities, and makes degradation projections. The current and voltage of selected array sections and the array switch status are monitored.

5.3.4 Power Source Management Processing Requirements

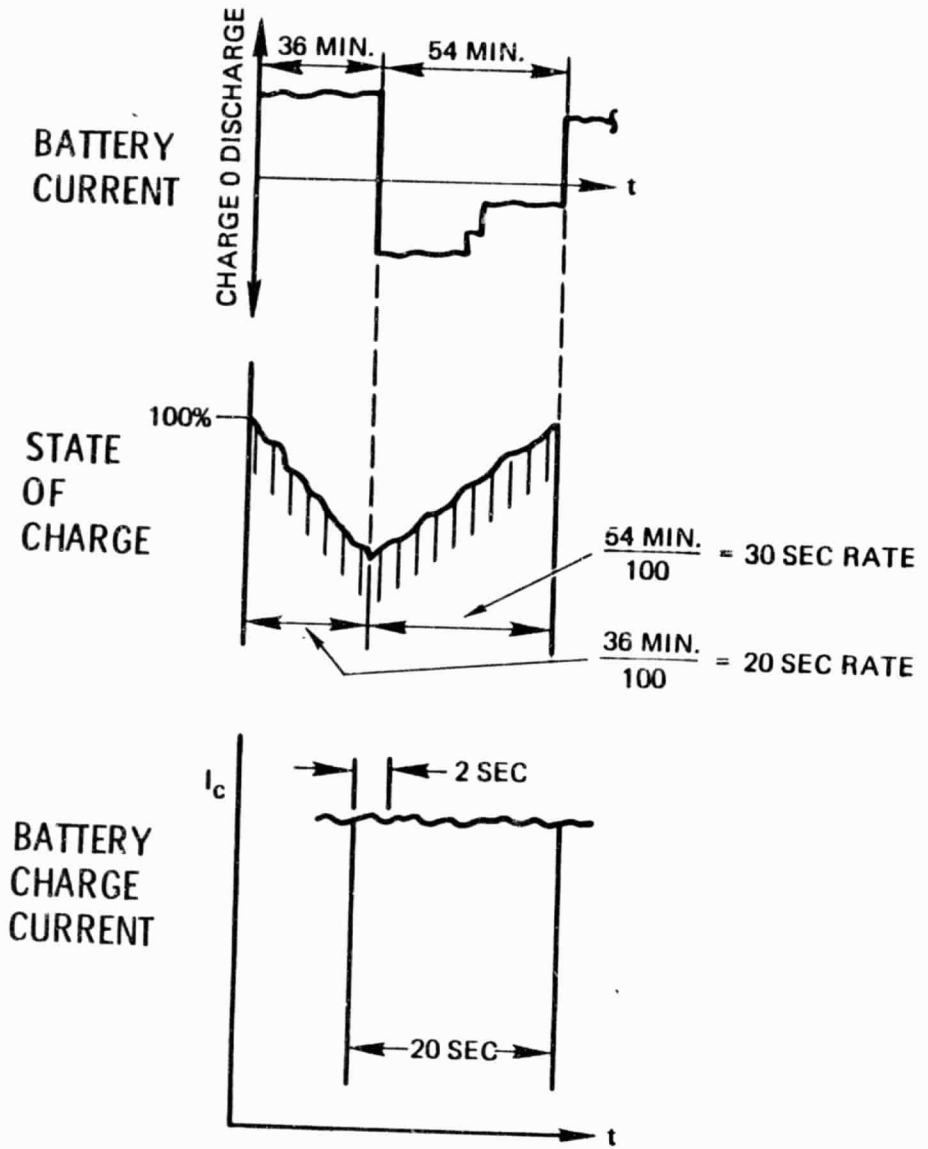
The power source management algorithms were evaluated to determine the computer processing requirements in terms of the number of instructions/second that are needed to perform the algorithms. This analysis required determining the number of times per minute that the algorithm must be repeated (monitor rate) and the total number of programming instructions required to define the algorithm. This analysis is summarized in Table 5-2.

Table 5-2. Power Source Management Processing Requirements

Function	Instructions per Function	Repetition Interval (sec)	Controller Instruction Rate (instr/sec)	System* Instruction Rate (instr/sec)
Battery State of Health	83,100	20	4,155	70,635
Battery Charge Control	7,100	2	3,550	60,350
Solar Array Status	3,500	20	175	2,975
			7,880	133,960

*Seventeen controllers in 250 kW system.

The monitor rates are based on the 90-minute orbit that was defined for the spacecraft (see Figure 5-6). The orbit is further defined as having a 36-minute eclipse period along with a 54-minute daylight period. Battery charging must be closely controlled during finish charge and trickle charge in order to avoid excessive overcharging or battery run-down. This trimming of the battery overcharge ratio (100 to 110 percent range) requires a state-of-charge integration accuracy of 1 percent over the charge and discharge cycles. The monitoring rate is therefore dictated by the eclipse period since it is the shortest period. The calculated monitoring rate requirement is then 36 minutes divided by 100 or approximately one sample every 20 seconds. In order for the measured parameter (battery current) not to inject additional error into the required



90-MINUTE ORBIT
DEFINES BATTERY CYCLE

TRIMMING BATTERY OVERCHARGE RATIO (100 - 110% RANGE) REQUIRES STATE OF CHARGE INTEGRATION ACCURACY OF 1%.

SMALLER INTERVALS NEEDED TO MAINTAIN CONTROL OF BATTERY CURRENT DURING FINISH AND TRICKLE CHARGING -USE 2 SECOND RATE-

FINISH CHARGE AND TRICKLE CHARGE CURRENTS MUST BE CLOSELY CONTROLLED TO AVOID EXCESSIVE OVERCHARGING OR BATTERY RUN DOWN

Figure 5-6. Battery Charge Control Requirements Define Monitor Rates

measurement accuracy, the monitoring rate of the battery current was increased by an order of magnitude over the 20-second requirement to one sample every 2 seconds.

The battery state of health and solar array status algorithms are performed every 20 seconds to accommodate the 1 percent accuracy requirement. Also battery cell monitoring at the end of the discharge cycle requires monitoring in the 20-second time frame to ensure that some cells are not completely discharged and reversed.

The number of instructions required to perform the algorithm were calculated based on past experience in programming similar algorithms for spacecraft applications.

5.4 LOAD CENTER MANAGEMENT ALGORITHMS

The load center management algorithms provide the processes and procedures for the functions required to operate and monitor the power distribution circuitry and the power conditioning circuitry (if required) that is located in the spacecraft load centers. The major load center management algorithms are command processing and switch and load bus monitoring. These algorithms are described in the following sections.

5.4.1 Command Processing

The command processing algorithm is initiated by the executive whenever a command is transmitted to the load center controller (LCC) by the EPS controller. The structured design of the command processing algorithm is shown in Figure 5-7 (see Appendix A for instructions on how to read the diagram). The incoming message is stored in an input buffer called INBUF. A message is sent to the LCC cathode ray tube (CRT) terminal indicating that the LCC has received a command. The message reads "COMMAND HAS BEEN RECEIVED FROM EPS." The message is then transferred from the input buffer to the message buffer (called MESBUF). The message is then examined to determine what type of command was sent from the EPS controller. Four types of commands are recognized by the LCC. These commands are load on/off switch commands, EPS data requests, EPS initialization requests, and circuit breaker programmable limit commands. The command formats are shown in Table 5-3. The incoming command is examined to determine which type of command has been sent from the EPS. If the command is not recognized in

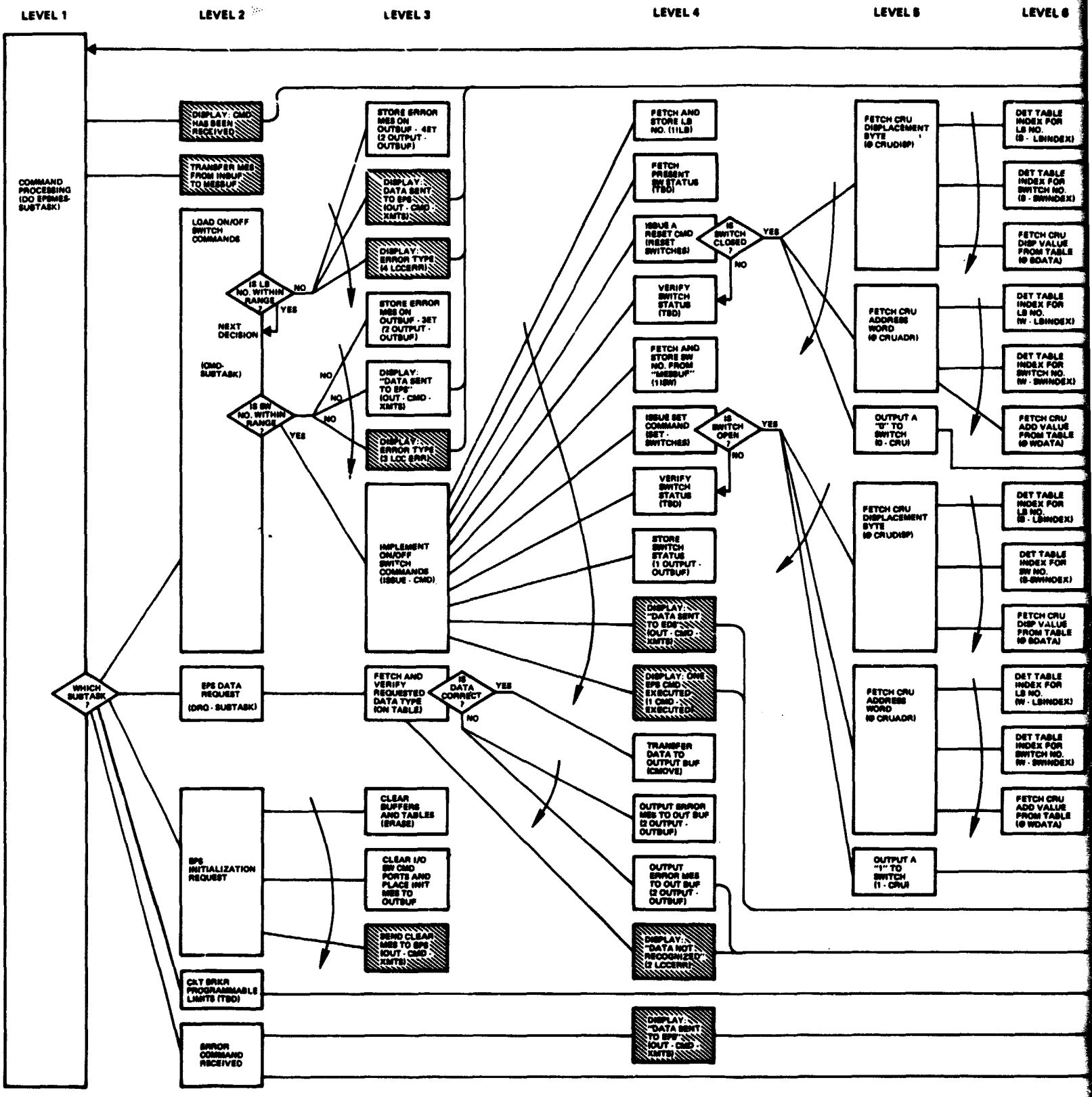


Figure 5-7. Load Center Co
Command Proces
Algorithm

FOLDOUT FRAME

ORIGINAL PAGE IS
OF POOR QUALITY

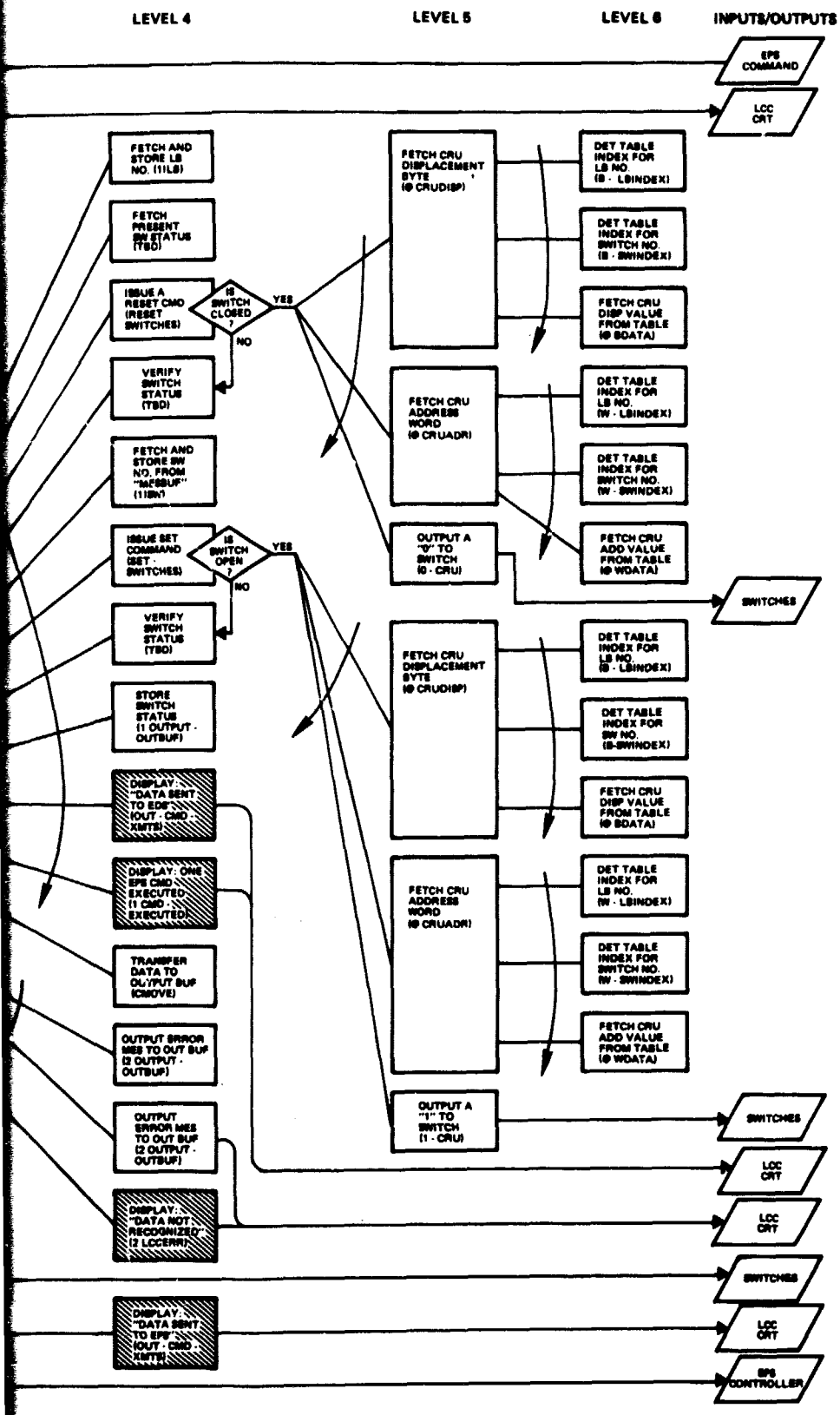


Figure 5-7. Load Center Controller Command Processing Algorithm

ORIGINAL PAGE IS OF POOR QUALITY

FOLDOUT FRAME

Table 5-3. Command Formats from EPS to LCC

Command Type	Command Neumonic (2 bytes)		Command Text (variable length)	Remarks
Load Command	1 CMD	XX*	Switch number; on/off command (2 bytes) (1 byte)	
Data Requests:			(none)	Requests are for data of all generic devices and hence require no message text to identify a specific device.
Load voltage	2DRQ	1LVD		
Switch current	2DRQ	2SCD		
Switch status	2DRQ	3SSD		
Channel current	2DRQ	4CMD		
CB trip setting	2DRQ	5LIM		
Initialization Command	3INT	XX*	(none)	Initializes all I/O and tables; no message text required
Command CB Trip Setting	4BKR	XX*	Breaker number; trip setting (2 bytes) (1 byte)	

*XX = Don't care

the command type table, the command is considered to be in error and an error message is sent to the EPS controller. Also a message is sent to the LCC cathode ray tube (CRT) which reads "CMD INVALID DATA HAS BEEN SENT TO THE EPS." The message formats from the LCC to the EPS are shown in Table 5-4. Each of the four command types are discussed in the following sections.

5.4.1.1 Load On/Off Switch Commands

The load on/off switch command algorithm issues the set and reset commands to the load bus circuit breakers in the power distribution network. Initially, the incoming command is checked to verify that the load bus number and the switch number are correct. If either is incorrect, an error message is placed in an output buffer and an error message is sent to the EPS controller (the error message is actually sent by the executive software). Also, a message is sent to the LCC CRT which reads "DATA HAS BEEN SENT TO THE EPS" and the error type is displayed on the LCC CRT.

If the incoming command is correct, then the switch on/off command is implemented. Initially the load bus number is read from the message buffer and its associated switch status is identified. If any of the switches are closed, a reset command is issued, which disconnects all power to the load bus. This procedure prevents two power channels from being tied together.* After the switch reset commands have been issued, the switch status is again verified. At this time, the switch number is read from the message buffer and the desired set command is issued. The switch status is again verified after this command is issued. The switch status information is placed in the output buffer and the verification data is transmitted to the EPS controller. Messages are sent to the LCC CRT that say "DATA SENT TO EPS" and "ONE EPS CMD EXECUTED."

*This procedure produces break before make switching. For special loads requiring uninterrupted power, the procedure may be reversed to produce make before break switching, but this requires reverse current prevention (diodes) in the switches to prevent interconnecting two power channels.

Table 5-4. Message Formats from LCC to EPS

Message Type	Message Neumonic (2 bytes)		Message Text (variable length)	Remarks
Switch Status	1SW-STATUS	XX*	Switch number, status (2 bytes) (1 byte)	Response to load command
Data:				
Load voltage	2DRQ	1LVD	Load number, voltage	Repeated for all loads
Switch current	2DRQ	2SCD	Switch number, current	Repeated for all switches
Switch status	2DRQ	3SSD	Switch number, status	Repeated for all switches
Channel current	2DRQ	4CHD	Channel number, current	Repeated for all channels
CB trip setting	2DRQ	5LIM	CB number, trip setting (2 bytes) (1 byte)	Repeated for all CBs
Errors:				
EPS command	3MES	1ET	Unrecognized command	EPS command type not recognized
Data type	3MES	2ET	Unrecognized data request	Requested data type not required
Switch number	3MES	3ET	Unrecognized switch number	Switch command number not recognized
Load number	3MES	4ET	Unrecognized load number	Load bus number not recognized
Data value	3MES	5ET	Data identifier, value	Analog/digital converter failed to provide end of conversion signal
Initialization Complete	4INI	XX*	(none)	I/O and table initialization is completed

*XX = Don't care

5.4.1.2 EPS Data Request

An EPS data request command is sent whenever the EPS controller requires data from the load center. Data requests for load voltage, switch current, switch status, channel current, and circuit breaker trip point levels are issued. The incoming data request is read and the requested data type is verified. If the data request is correct the data is transferred to the output buffer and the executive software will send the data to the EPS controller. If the data request is incorrect, an error message is sent to the EPS controller and a message is displayed on the LCC CRT that states "DATA NOT RECOGNIZED." Also, "DATA SENT TO EPS" is displayed on the LCC CRT terminal.

5.4.1.3 EPS Initialization Request

The EPS initialization request is issued during any start-up period after a power outage. The initialization request clears all buffers and tables and input/output switch command ports. After the initialization request has been implemented, a message is sent to the EPS controller.

5.4.1.4 Circuit Breaker Programmable Limits

The circuit breaker programmable limits commands are issued when the trip points on the load bus circuit breakers need to be adjusted. The structured design of this algorithm will generally follow that of the other command algorithms. Initially, the incoming command will be verified. If an incorrect command is detected, a message will be sent to the EPS controller and a message will be displayed on the LCC CRT. If a correct command is detected, then the command will be implemented and a message will be displayed on the LCC CRT.

5.4.2 Switch and Load Bus Monitoring

The switch and load bus monitoring algorithm consists of the monitor algorithm and the fault interrogation algorithm. The monitor algorithm collects and assembles the power distribution network telemetry and status data, and the fault interrogation algorithm analyzes the data to determine if failures have occurred in the power distribution network. These algorithms are described below.

5.4.2.1 Monitor Algorithm

The monitor algorithm is a routine process that is activated periodically by the LCC executive software. The structured design of the monitor algorithm is shown in Figure 5-8. (See Appendix A for instructions on how to read the diagram.) Five types of distribution network data are monitored: load voltages, channel currents, switch currents, switch status, and circuit breaker trip-point limits. The current and voltage monitoring procedures are generally alike. The address of the input port to be monitored is read from a table and the port is activated. An analog-to-digital converter located in the load center controller is activated and sends an end-of-conversion (EOC) signal when it is finished. The algorithm verifies the EOC signal. If the EOC signal is found, the telemetry data is stored in a table. If the EOC signal is not found, an error message is placed in the LCC output buffer and a "DATA SENT TO EPS" message is displayed on the LCC CRT. Also, an error message is displayed on the LCC CRT that states "ADC FAILED TO RETURN EOC."

The switch status monitor algorithm uses a status byte to record the switch status. Initially, the status byte is set to zero. Then the switch status is read through the appropriate input port. If the switch is closed, a one is placed in the appropriate bit in the status byte. If the switch is open, the status byte remains at zero.

The circuit breaker trip-point limits have not been programmed at this time.

5.4.2.2 Fault Interrogation Algorithm

The fault interrogation algorithm interrogates the status of the switchgear and the electrical parameters of the respective load buses (or terminals) to determine compliance with the last received command or to identify the failure mode. The last switch command, the present switch status, and the load current and voltage are monitored, and the data stored. This data is compared to the tabulated set of conceivable circumstances (Table 5-5). A status condition is generated (Table 5-5, Column 5), and the related message is sent to the EPS controller. If a failure is indicated (Fault Types 1 through 12), the load priority table applicable to

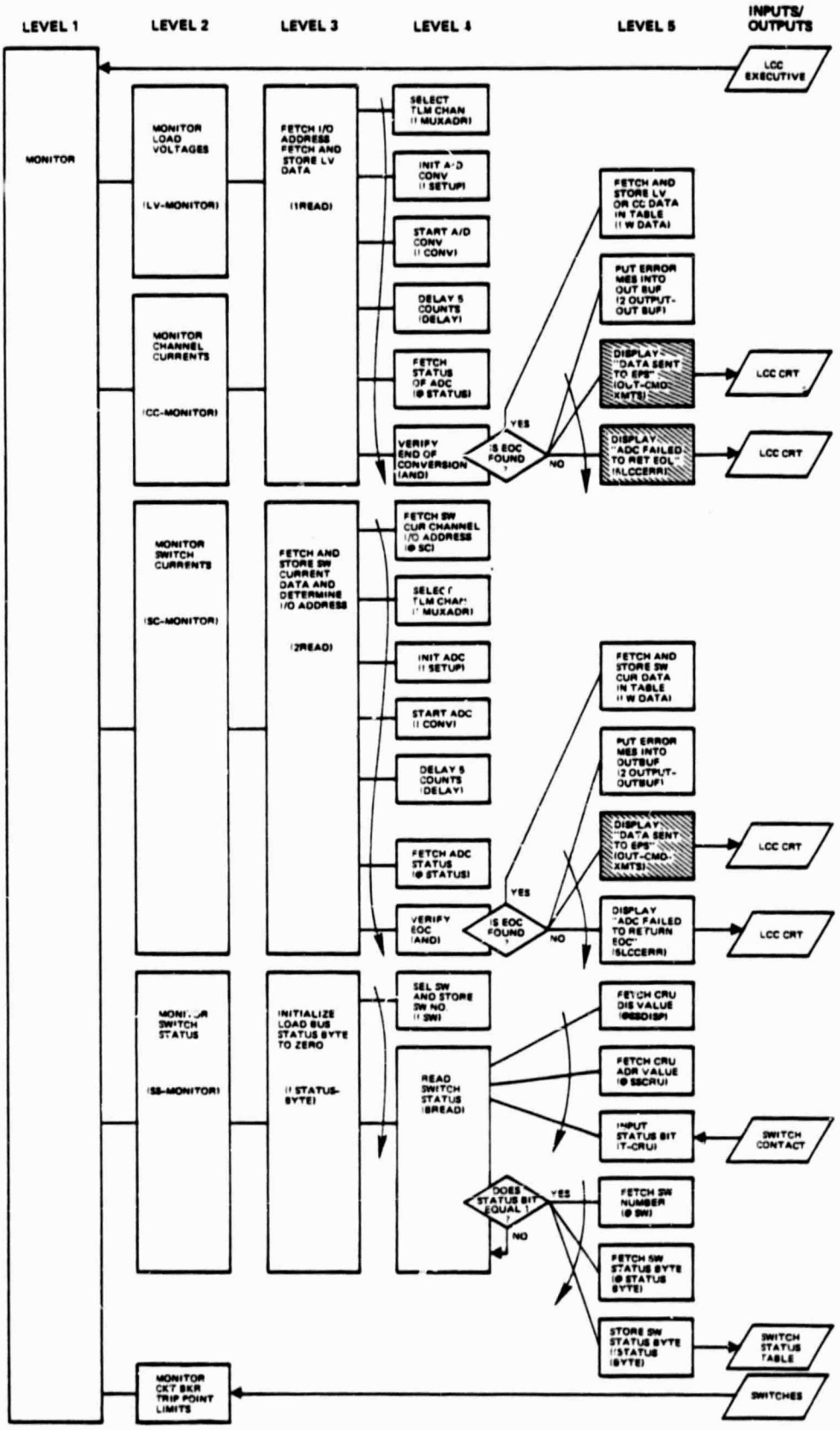


Figure 5-8. Load Center Controller Monitor Algorithm

Table 5-5. Status Table

Last Command	Load-Bus Current	Load Bus Voltage	Switch Status	Status Type	Status Condition	EPS Controller Reaction
On	Normal	Normal	On	0	No fault; load on	Bus on; OK
Off	0	0	Off	0	No fault; switch off	Bus off; OK
On	0	0	Off	1	Switch open	Change load table connection options
Off	Normal	Normal	On	2	Switch closed	Change load table connection options
On	Normal	Normal	Off	3	Switch status open	Disable validation check
Off	0	0	On	4	Switch status closed	Disable validation check
On	0	Normal	On	5	Current transducer, no output	Verify current from main bus data; flag transducer failure
On	Maximum	Normal	On	6	Current transducer, full output	Verify current from main bus data; flag transducer failure
On	$<I_L$ Min	Normal	On	7	Load undercurrent	Load problem alarm to spacecraft
On	$<I_L$ Max	Normal	On	8	Load overcurrent	Load fault; alarm to spacecraft
On	Normal	0	On	9	Voltage transducer, no output	Verify load performance; flag transducer failure
On	Normal	Maximum	On	10	Voltage transducer, full output	Verify load performance; flag transducer failure
On	Normal	$<V_L$ Min	On	11	Bus undervoltage	Undervoltage alarm; reassign loads
On	Normal	$>V_L$ Max	On	12	Bus overvoltage	Overvoltage alarm; reduce array output

the bus assignment algorithm is modified to indicate the new constraint on the bus assignment options of that load bus/terminal.

Two significant assumptions are made in developing the status table and the resulting failure conditions and EPS controller messages. First, data processing and storage failures are neglected. A redundancy study is required to define the appropriate accommodation approaches for such failures. Second, only a single component failure is considered. Concurrent multiple failures significantly expand the status table. However, the corrective action taken after the initial failure, particularly a switch failure, usually preempts the effects of further failures. Hence, this approach is adequate, but the final algorithm must preclude misoperation with multiple failures. A simple memory approach that retains the fact of a switch failure regardless of subsequent failures will be considered.

Loads are distributed throughout the power subsystem distribution network, but their associated switchgear is geographically grouped into load centers for control purposes. Each load center controller implements the switch commands and monitors the resulting performance of the commanded distribution switchgear. Hence, each load center controller incorporates this algorithm and executes it independently of other controllers.

5.4.3 Load Center Management Processing Requirements

The load center management algorithms were analyzed to determine the computer processing requirements in terms of the number of instructions per second that are needed to perform the algorithms. This analysis required determining the number of times per minute that the algorithm must be repeated and the total number of programming instructions required to define the algorithm. This analysis is summarized in Table 5-6.

A repeat time of one sample every 2 seconds was selected for both load center management algorithms. This time was based on an engineering judgment of the time that should be required to report or to execute a change in the state of the electrical power subsystem. For instance, during the load bus assignments procedure, a large number of switch on/off commands are issued to and executed by the load center controller. Each command is verified, switch status is monitored, and a verification signal is sent

Table 5-6. Load Center Management Processing Requirements

Functions	Instructions per Function	Repetition Interval (sec)	Controller Instruction Rate (instr/sec)	System* Instruction Rate (instr/sec)
Switch and load monitoring and fault interrogation	17,000	2	8,500	85,000
Load on/off command processing	2,600	2	1,300	13,000
			9,800	98,000

*Ten controllers in 250 kW system.

back to the EPS controller. This process must be performed rapidly in order for the controllers to perform other tasks.

The number of instructions required to perform the algorithm was calculated based on past experience in programming similar algorithms for spacecraft applications.

5.5 EPS MANAGEMENT ALGORITHMS

The EPS management algorithms provide the processes and procedures for the overall power subsystem level functions. The major EPS management algorithms are energy planning and allocation, load bus assignments, and power subsystem state of health. These algorithms are described in the following sections.

5.5.1 Energy Planning and Allocation

The energy planning algorithm provides the comprehensive load management to assure adequate power and energy margins for each power channel. This energy management function includes the analysis and planning to utilize effectively the full capabilities of the electrical power sources (solar array segments and batteries) and to enhance the life of the batteries. These objectives are accomplished by allocating judicious loading to each power source channel in order to attain balanced (or otherwise planned) operation of normal batteries and lighter loading of degraded batteries.

The energy planning algorithm utilizes data from the battery charge control, and battery state-of-health, solar array status, and power subsystem state-of-health algorithms. This data is transmitted from the power source and load center controllers to the EPS controller and stored in local memory for ready access.

The energy planning algorithm (Figure 5-9) has two major branches: one for sunlight operation and one for eclipse operation. During eclipse the state of charge of each battery is calculated, the depth of discharge at the end of eclipse projected, and the channel-to-channel imbalance estimated based on projected battery depth of discharge. If the imbalance is too great, load reassignment is requested of the bus assignment algorithm based upon the desired new load levels calculated by the energy planning algorithm for each channel.

In some circumstances the aggregate load may project a system depth of discharge exceeding the nominal operational limit (e.g., 33 percent). The spacecraft control authority (computer and ground) is then notified requesting a reduction in load. However, this overload is supported without restraint until a second, more severe system depth-of-discharge limit is projected to be exceeded at the end of eclipse (e.g., 50 percent) unless a load reallocation command is received. With this overload condition, the spacecraft control authority is notified of this second phase, and load buses of low priority are deactivated (shut down) to the extent available, or until the aggregate load projects an acceptable system depth of discharge at the end of eclipse (e.g., $\leq$33 percent). With this change in loads, the bus assignments are recomputed by the bus assignment algorithm.

Priority constraints may preclude significant unloading in this second phase. Should a third level of system depth of discharge actually be exceeded (e.g., 65 percent), an emergency notice is sent to the spacecraft control authority, and load buses are disabled in reverse order of priority, but in direct order of load size within a priority category, until an acceptable system load level is attained. This load level, if continued to the end of eclipse and added to the existing system depth of discharge, must not exceed an acceptable one-time battery depth-of-discharge utilization (e.g., 80 percent). Again, bus assignments are recomputed by the bus assignment algorithm. In addition, one or two subsequent sunlight periods

Figure 5-9. Energy Planning Algorithm Flow Diagram

with reduced load may be required to fully recharge the batteries after such an incident.

Terrestrial modification of these results is possible by programming a new set of limits for system depth of discharge; for example, 40, 55, 70 instead of the suggested 33, 50, 65 examples. The major objective is to support loads, even though temporarily excessive, until the last possible moment before curtailment of operation is necessary for survival. Such curtailment is abnormal, and only due to extreme overloads or an unusual degree of degradation.

During sunlight operation, the energy planning algorithm calculates the available system energy for battery charging and projects the state of charge at the beginning of the next eclipse. If full recharge is attainable, channel load balancing based upon existing state of charge is calculated. Excessive imbalance, leading to deficient charging for one or more batteries, generates a request of the bus assignment algorithm to reassign loads based upon the newly calculated desired channel loadings.

A projection of insufficient energy for full system recharge leads to two alternatives: (1) continue operation without assurance of attaining full recharge or (2) enter a mandatory load reduction routine to make sufficient energy available to fully recharge the batteries. Mandatory load reduction occurs when the system state of charge is projected to fall below a safe value (e.g., 50 percent), or if full recharge has not been achieved recently (e.g., over 15 eclipse cycles, approximately one day). The spacecraft control authority is notified of the pending action, and sufficient load buses are deactivated (shut down) with appropriate priority and load value considerations until full recharge capability is projected.

Operational load support is continued if the system state of charge is above the safe value (e.g., 50 percent) and the system has been recently fully charged (e.g., less than 16 eclipses, or less than a day). When the spacecraft control authority is requested to reduce loads, the desired channel load levels are recalculated by the energy planning algorithm, and the load buses are reassigned by the bus assignment algorithm to attain a reasonably uniform state of charge (or other preplanned distribution) on the batteries.

Terrestrial modification of these results is possible by programming a new set of limits; for example 60 percent and 4 and 5 eclipses instead of 50 percent and 15 and 16 eclipses. Again, the major objective is to support loads, even though temporarily excessive, as long as possible before curtailment of operation.

5.5.2 Load Bus Assignments

The configuration of the distribution system (load center switch selections) is dependent upon a large and complex interaction of parameters (Figure 5-10). The central control algorithm for definition of the distribution system configuration is the load bus assignment algorithm. The load bus assignment algorithm is supported by other algorithms and data tables: energy planning algorithm, load command processing, switch monitor algorithm, state-of-health algorithm, and various load (or load bus) data tables, including switch options, initial load currents, latest load currents, switch failures, load priorities, and power quality desires, etc.

The load bus assignment algorithm, as the name implies, provides a logical assignment of loads (or load buses) to the power channels (main buses) and as such defines the distribution configuration of the electrical power system. The output of this algorithm is a table of distribution switch closure assignments that identify the computer selected distribution configuration. To define the desirable distribution configuration, the load bus assignment algorithm evaluates the desired power channel loading, the existing channel loading, existing load currents, expected currents for new loads, the specified operational profile, and the distribution switching options. The desired channel loading data comes from the energy planning algorithm. The expected load currents and the load-to-channel switching options are provided by the load priority table. The desired load implementation (load operational pattern) is provided by the spacecraft controller or from terrestrial commands.

5.5.2.1 Load Bus Assignments Methodology

One method of selecting the best configuration for the distribution system is to analyze all of the possible configurations, determine the mismatch of each channel for each configuration, and select the one configuation which minimizes the imbalance among the channels. However, this

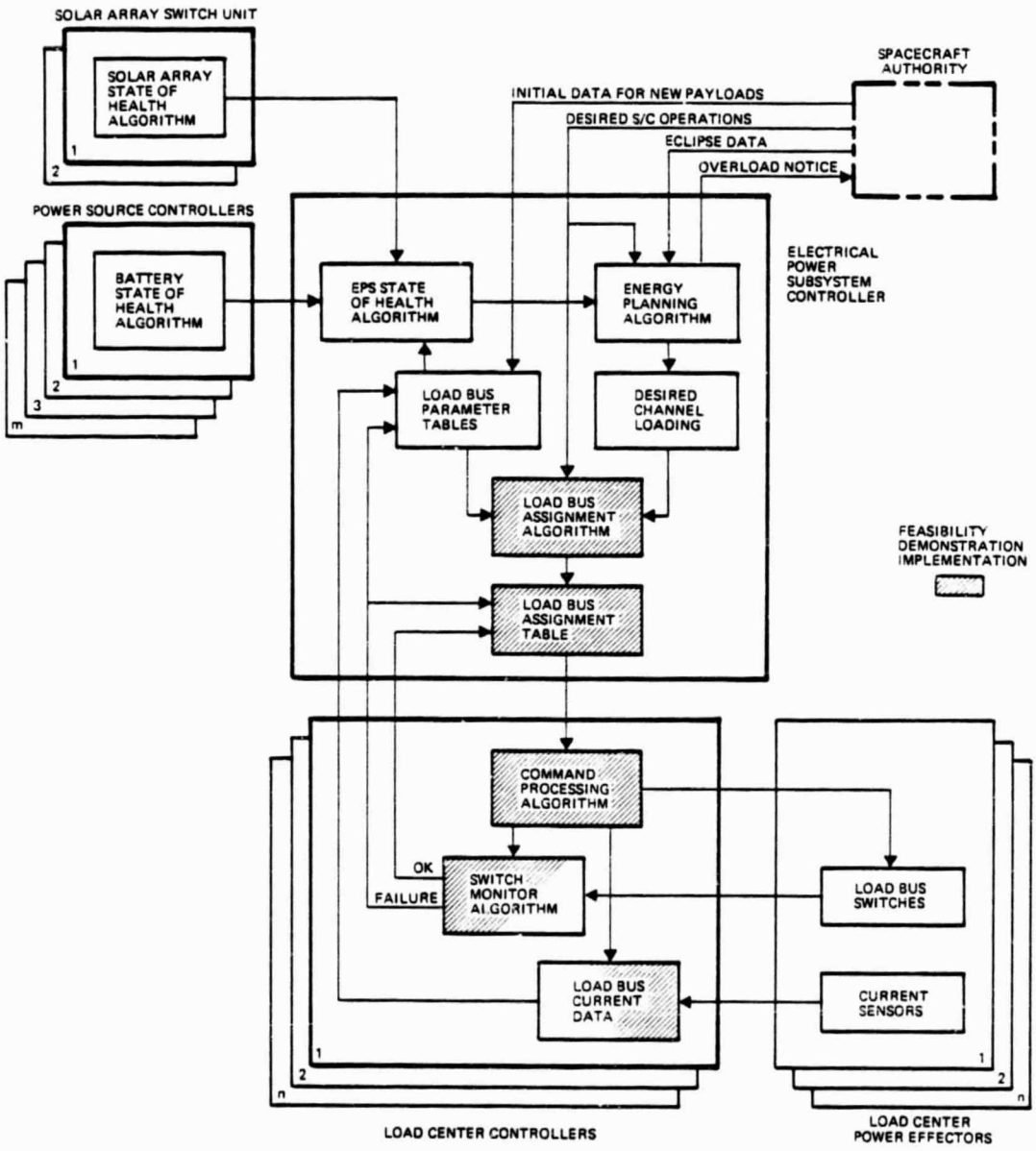


Figure 5-10. Distribution Configuration Process - Load Bus Assignment

approach is impractical. The total number of possible distribution configurations is given by k^n where n is the number of loads or load buses and k is the number of possible channel connections for each load (or load bus). On a 250-kilowatt power system, over 200 load buses are anticipated with a minimum of two channel connections per load bus. A total of 1.6×10^{60} distribution configurations would need evaluation. With an extremely optimistic computer speed where one configuration is evaluated every microsecond, this solution approach would require 5×10^{46} years to compute!

A practical approach to load assignment is attained by ranking the load (load buses) by their respective power (current) and then segregating this load list into "g" groups of "N" loads according to their relative power, largest loads in the first group, smallest loads in group "g." All the distribution configurations for the first load group (highest power) are computer evaluated, and the configuration for minimum channel imbalance is selected. The channel loading results for this first load group are subtracted from the desired channel loading, and the results become the desired channel loading for the second group analysis. This procedure is repeated for each successively lower powered load group until all groups have been evaluated.

This approach significantly reduces the number of configurations for evaluation when small group sizings are employed. The total number of configurations evaluated employing this approach is:

$$\sum_{i=1}^{g} K_i^{N_i} = \text{number of configurations}$$

where

g = the number of load groups

N_i = the number of load buses in a given group

K_i = the number of channel connection options available for each load bus in the group

For a group sizing equal to 10 load buses (N = 10), the total number of groups becomes 20 (g = 20) for the 250-kilowatt system with 200 load buses.

The total number of configurations evaluated is then reduced to 20.4×10^3 and these can easily be evaluated in several minutes.

The result of this grouped approach may not consider the specific configuration that produces absolute minimum unbalance. However, within the constraint of the available load increments, a very low unbalance and readily acceptable configuration is attained with reasonable computation time. Hence, a cost-effective solution is attained, though not necessarily the absolute minimum unbalance that it might be possible to attain.

5.5.2.2 Load Bus Assignment Priorities and Constraints

The load bus assignment algorithm must accommodate certain operational priorities and constraints imposed by the spacecraft system. Examples of such criteria that would affect the assignment of load buses to channels are listed below:

a) In general, the load bus assignment algorithm will balance the load on each power channel (see Trade Studies and Analyses, Section 8). However, in certain cases when power sources are degraded, the loads will be unbalanced based on a specified load ratio.

b) One or more "quiet" channels may be designated for sensitive loads

c) One or more channels may be designated for particularly noisy or pulsed power loads

d) Limitations may be applied to the switching of loads such as "make before break" to assure continuous power availability, or some loads may be designated as "undesirable" to switch

e) Load priorities may be established for normal, emergency, start-up, and other specific operating conditions such as limited power availability

f) Criteria for rebalancing the channel loads must be established based on a specified sensitivity to the amount of allowable unbalance per channel. Also, sensitivity to pulsed loads must be established when determining the rebalance criteria.

All of these criteria (and/or others) may be employed in defining the assignment of loads to power channels.

5.5.2.3 Structured Design of the Load Bus Assignments Demonstration Algorithm

A demonstration algorithm was developed for the load bus assignments function in order to verify the feasibility of the above-mentioned computational concepts. A power distribution network configuration was assumed, as shown in Figure 5-11. Load priority tables were established for load bus current, channel connections, switch positions, and system load priorities, as described in the power subsystem state-of-health algorithm, Section 5.5.3. (The load priority tables are a part of the power subsystem state-of-health algorithm.) A generalized algorithm was developed and a specific example was performed to verify functional operation. The data that was entered into the load priority tables is shown in Table 5-7. Each load bus was assigned a relative bus number which is used by the computer to identify the bus. Load bus currents were arbitrarily assigned as shown in the load bus current file (LC-LPT). The channel connections reflect the configuration of Figure 5-11. The switch positions are shown in the initial condition that is assumed when the configuration selection routine is performed. (A one indicates that the switch is closed and a zero indicates that the switch is open.) All the loads were assumed to be switchable (Priority 3) for this demonstration algorithm. The algorithm assumes that the loads will be balanced evenly across the three power channels. In other words, the desired channel loading (DCL) is the sum of the load currents divided by three. (Later on, when a complete power management system is operating, the desired channel loading will be determined by the energy planning process.)

This demonstration algorithm is a building block to an operationally viable algorithm. The demonstration algorithm requires expansion to incorporate further operational priorities and power quality constraints and to receive variable channel loading inputs from the energy planning algorithm. The expansion of this basic demonstration algorithm is projected for a future algorithm development contract.

A structured diagram of the load bus assignments algorithm is shown in Figures 5-12 and 5-13. (An explanation of how to read the structured diagram is presented in Appendix A.) Seven main modules are used to describe the algorithm. To begin, any new load buses which need to be activated are

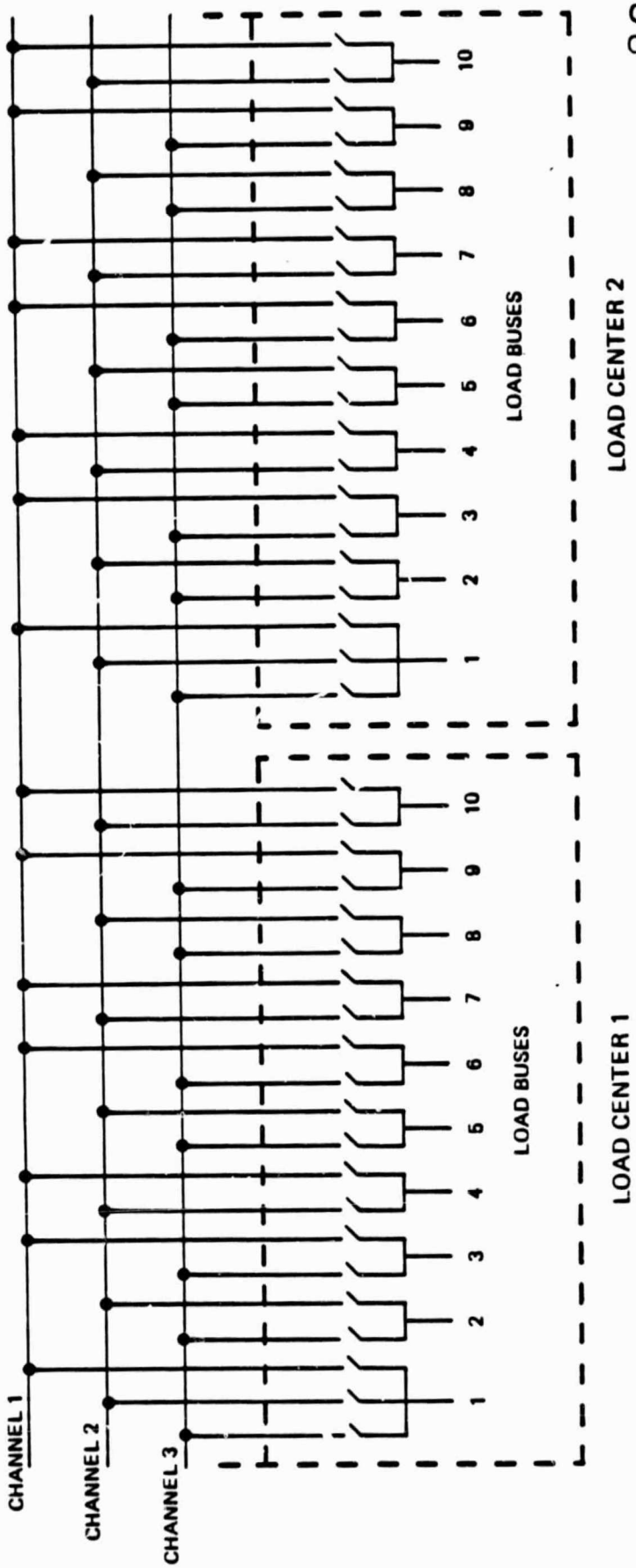
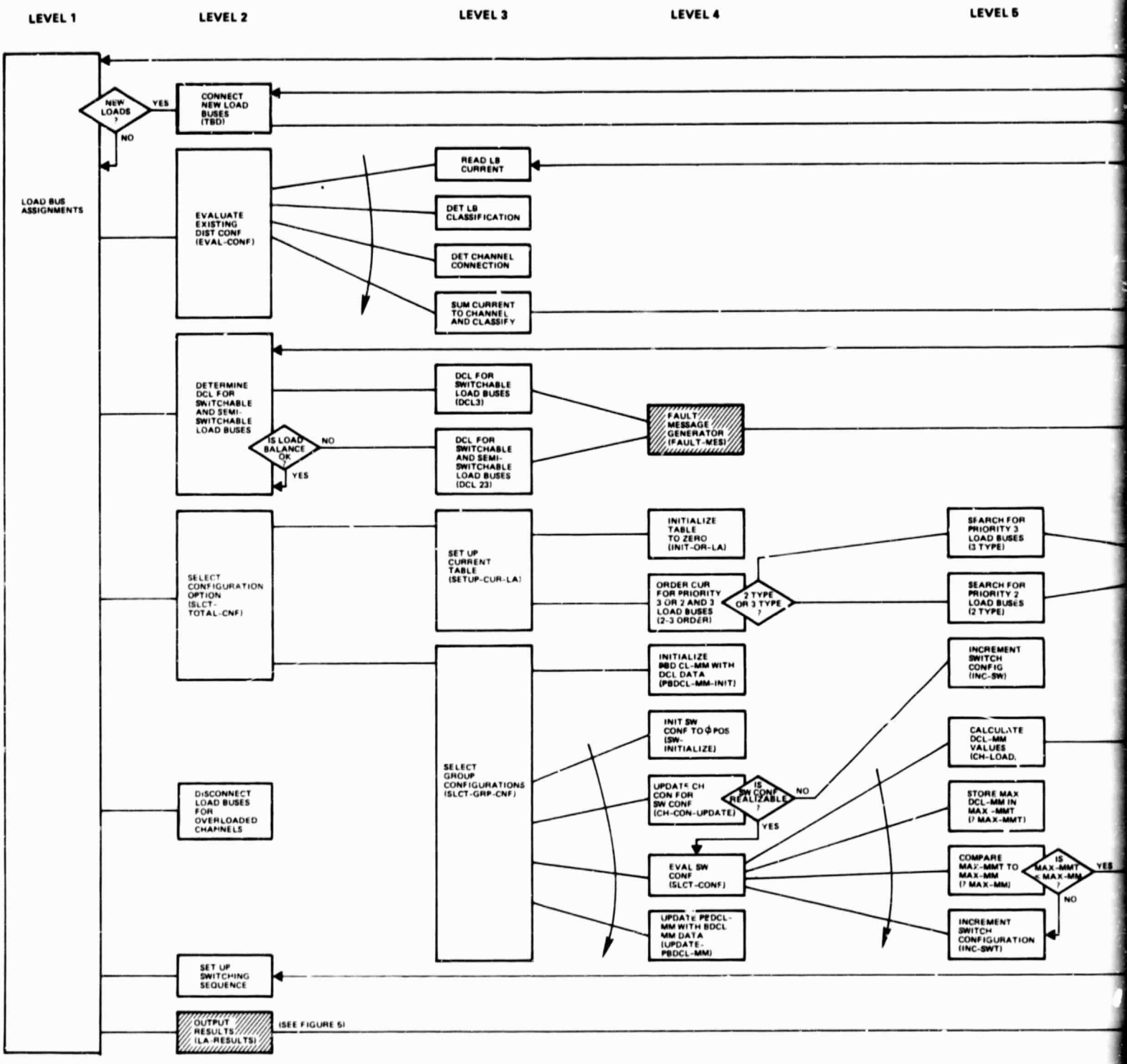


Figure 5-11. Power Distribution Network for Load Bus Assignments Demonstration Algorithm

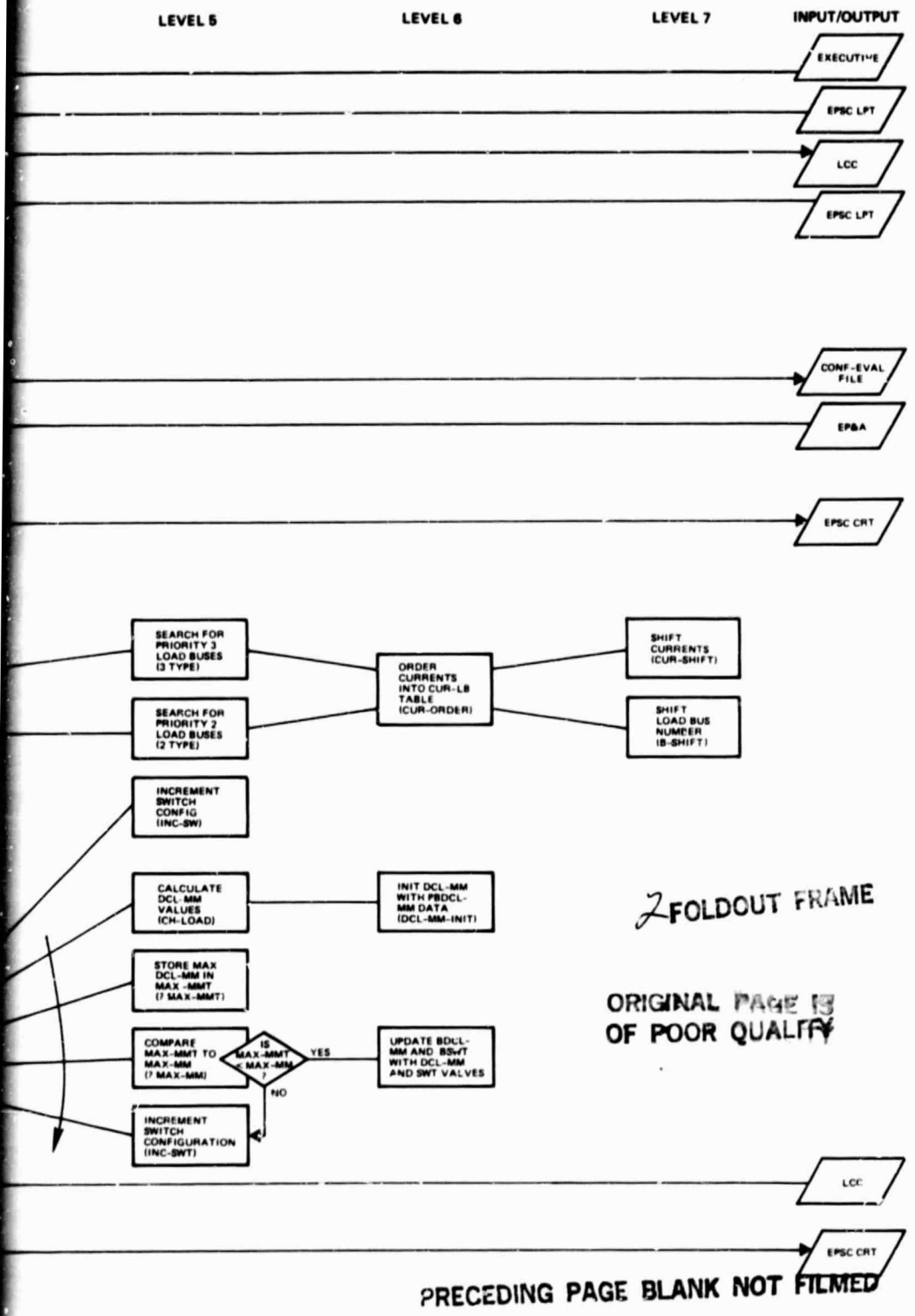
Table 5-7. Load Priority Table Data for Load Bus Assignment Demonstration Algorithm

Load Bus No.	Relative Bus No.	Load Bus Current (amps) (LC-LPT)	Channel Connections (LHC-LPT)			Switch Positions (SW-LPT)			System Load Priorities (SL-LPT)	
			SW 1	SW 2	SW 3	SW 1	SW 2	SW 3	Norm	Emer
1-1	1	10	Ch 1	Ch 2	Ch 3	1	0	0	3	3
1-2	2	15	Ch 1	Ch 2	0	1	0	0	3	3
1-3	3	15	Ch 1	Ch 3	0	1	0	0	3	3
1-4	4	15	Ch 2	Ch 3	0	1	0	0	3	3
1-5	5	7.5	Ch 1	Ch 2	0	1	0	0	3	3
1-6	6	7.5	Ch 1	Ch 3	0	1	0	0	3	3
1-7	7	7.5	Ch 2	Ch 3	0	1	0	0	3	3
1-8	8	3.75	Ch 1	Ch 2	0	1	0	0	3	3
1-9	9	3.75	Ch 1	Ch 3	0	1	0	0	3	3
1-10	10	3.75	Ch 2	Ch 3	0	1	0	0	3	3
2-1	11	10	Ch 1	Ch 2	Ch 3	1	0	0	3	3
2-2	12	15	Ch 1	Ch 2	0	1	0	0	3	3
2-3	13	15	Ch 1	Ch 3	0	1	0	0	3	3
2-4	14	15	Ch 2	Ch 3	0	1	0	0	3	3
2-5	15	7.5	Ch 1	Ch 2	0	1	0	0	3	3
2-6	16	7.5	Ch 1	Ch 3	0	1	0	0	3	3
2-7	17	7.5	Ch 2	Ch 3	0	1	0	0	3	3
2-8	18	3.75	Ch 1	Ch 2	0	1	0	0	3	3
2-9	19	3.75	Ch 1	Ch 3	0	1	0	0	3	3
2-10	20	3.75	Ch 2	Ch 3	0	1	0	0	3	3



ORIGINAL PAGE IS
OF POOR QUALITY

FOLDOUT FRAME



FOLDOUT FRAME

ORIGINAL PAGE IS OF POOR QUALITY

PRECEDING PAGE BLANK NOT FILMED

Figure 5-12. Load Bus Assignments Algorithm

PAGE 84 INTENTIONALLY BLANK

Figure 5-13. Load Assignment Routine

selectively connected to an appropriate channel, avoiding possible overloading of any of the power channels; then the existing distribution system configuration is evaluated to determine the channel loading conditions. Using this data the desired channel loading (DCL) for the switchable loads (and the semiswitchable loads in some cases) is calculated. The configuration options are then evaluated to determine the best configuration. If there are overloading conditions which cannot be resolved by reconfiguration, some load buses will be selectively disconnected using their emergency load bus priorities. The switching sequence is then set up, the commands are transmitted and the resulting configuration is displayed on the electrical power subsystem controller terminal. Each of these functions are discussed in subsequent sections.

5.5.2.3.1 Connect New Load Buses

This module will check to see if any new load buses are to be powered up. If there are, it will determine to which channel the load bus is to connect. This process will be supported by the power subsystem state-of-health algorithm which will expand its load priority table to account for the new load buses (see Section 5.5.3). Once it has been determined that new loads are to be serviced by looking at the initializing status in the load priority table, the distribution buses are powered up in the order given by the start-up priorities. The start-up priorities are used to assure that any dependence among the loads are accounted for (e.g., the dependence of some experiments on experiment support equipment). The proposed priority method will allow a group of load buses to be turned on and tested before further load buses are initialized. This is handled by identifying a group of load buses by a common start-up priority number. Starting with the lowest priority, the loads in the group are turned on one at a time, sequentially as they are listed in the load priority table. Then the load bus testing which is required is implemented before the next group of load buses is turned on. The channel connection chosen for each load bus is based on the maximum power margin for the connections available to the load. The power margin in this context refers to the difference between the present channel loading and the maximum load limit. Once all of the new load buses are connected, the energy planning and allocation is reevaluated with the new loads on the system.

5.5.2.3.2 Evaluate the Present Distribution System Configuration

The load assignment function sets up the configuration-evaluation (CONF-EVAL) file which stores the channel currents for each channel.

The channel currents are subdivided, and categorized by their channel number and the system load priorities as described. The CONF-EVAL file is set up as shown in Figure 5-14 with the total load current for each channel for each priority of 0, 1, 2, 3, and 4 stored independently. Note that all the channel currents are stored as double precision words (32 bits). The total channel load current is stored in the eleventh and twelfth 16-bit word locations allotted for each channel.

The FORTH word defined to assemble this data is EVAL-CONF. When EVAL-CONF is invoked the CONF-EVAL file is first initialized to zero. Then taking one bus at a time the load current is read from the LC-LPT file, its classification (and the channel which is supplying it) is determined, and the current is summed to the appropriate current in the CONF-EVAL file. Once all the load bus currents are summed into their designated memory locations, the currents supplied by each channel are totaled by summing the currents of the five classifications and this value is stored.

The data in the CONF-EVAL file in Figure 5-14 reflects the load currents and switch positions of Figure 5-17. Since we have assumed that all the load buses are Priority 3, the total channel current coincides with the Priority 3 channel current. (This configuration is clearly not the desired one, but illustrates the algorithm.)

5.5.2.3.3 Determine the Desired Channel Loads for the Switchable (and Semiswitchable) Load Buses

The load assignments algorithm calculates the desired channel loading (DCL) for each of the channels for the switchable loads (i.e., Priority 3). The algorithm was developed such that it will also calculate the desired channel loading for both the switchable and semiswitchable loads, i.e., Priorities 2 and 3, if it is desired. (This would require reassignment of some of the load buses to Priority 2 in the system load priorities table.) Hypothetically, the request for priority 2 and 3 loads would be made if no satisfactory loading conditions resulted from the configurations evaluated for Priority 3 loads only.

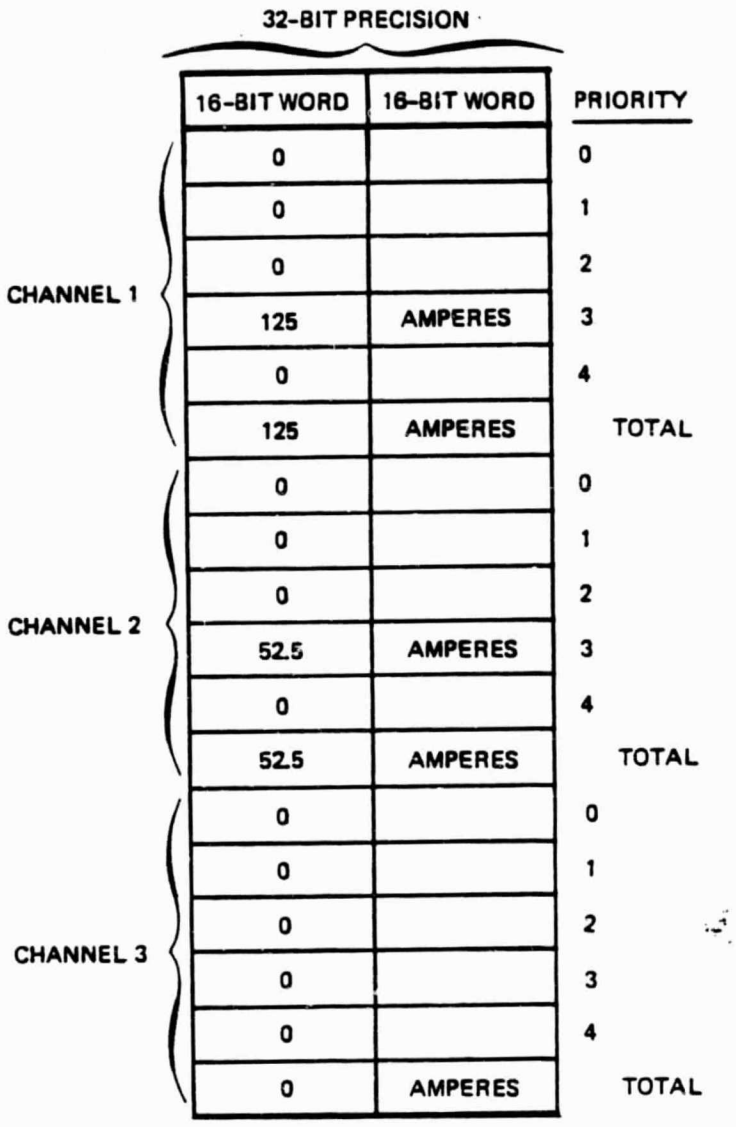


Figure 5-14. Configuration Evaluation (CONF-EVAL) File Storage in Memory

The FORTH word defined to calculate the desired channel loading for the Priority 3 loads is DCL3. The inputs to the routine are the DCLs which are the output of the energy planning and allocation. The resulting output of DCL3 is the channel current which should be made up of the Priority 3 loads. The DCLs resulting from the energy planning and allocation algorithm give the total load current for each channel which is made up of all five load priorities. Therefore, to obtain DCL3, the load currents for Priorities 0, 1, 2, and 4 must be subtracted from the DCL for each channel. Since these current values are stored in the CONF-EVAL file along with the Priority 3 currents and the total channel current the DCL3 can most quickly be determined for each channel by the following calculation:

$$DCL3_i = DCL_i - \text{Total Channel Load}_i + \text{Priority 3 Channel Load}_i$$

where i = one to three channels for the demonstration algorithm

If the resulting DCL3 is less than zero a fault message will be generated. This fault message has not been programmed at this time.

Continuing the example, since the Priority 3 load buses are the total load on each channel, DCL3 = DCL for each channel. As previously discussed, the desired channel loading is equal to the sum of all the loads divided by the number of channels (three). Therefore, DCL3 = 59.2 amperes for each channel.

5.5.2.3.4 Select Configuration Option (SLCT-TOTAL-CNF)

This process sorts the load bus currents according to magnitude and then evaluates the configuration options so that the selection of the best configurations is made. The process sets up a new set of tables, as shown in Table 5-8, that are constructed from the load priority tables of Table 5-7.

5.5.2.3.4.1 Set Up Current Table for Load Bus Assignments (SETUP-CUR-LA)

This process sets up a load bus current table (CUR-LA) along with the associated relative bus number (B-LA) table by sorting through all of the load bus currents one at a time and then arranging the currents for the switchable (and semiswitchable, if desired) load buses in descending order

Table 5-8. Configuration Option Selection Load Bus Assignment Table

Group No.	Load Bus Current (CUR-LA)	Relative Bus No. (B-LA)	Load Bus No.	Switch Configuration (SWT)	Channel Connection (CH-CON)	Best Switch Configuration (BSWT)
1	15	2	1-2	0	1	0
	15	3	1-3	0	1	0
	15	4	1-4	0	2	0
	15	12	2-2	0	1	0
	15	13	2-3	0	1	0
2	15	14	2-4	0	2	0
	10	1	1-1	0	1	0
	10	11	2-1	0	1	0
	7.5	5	1-5	0	1	0
	7.5	6	1-6	0	1	0
3	7.5	7	1-7	0	2	0
	7.5	15	2-5	0	1	0
	7.5	16	2-6	0	1	0
	7.5	17	2-7	0	2	0
	3.75	8	1-8	0	1	0
4	3.75	9	1-9	0	1	0
	3.75	10	1-10	0	2	0
	3.75	18	2-8	0	1	0
	3.75	19	2-9	0	1	0
	3.75	20	2-10	0	2	0

starting with the largest current magnitude first. The relative bus number is stored "adjacent" to its respective current for accounting purposes. The data in the tables of Table 5-8 show the resulting ordered currents for our specific example. The software modules which support this process are shown in Figure 5-12.

5.5.2.3.4.2 Select Group Configurations Independently (SLCT-GRP-CNF)

The process of stepping through each of the configurations and evaluating that configuration for the best balanced condition is programmed by this module. The load buses are evaluated in groups of 5, as shown in Table 5-8, starting with the group that has the largest current magnitudes. There are two main levels of software modules for this selection process. The first level (Level 4 of Figure 5-12) sets up the parameters for the group and then the second level (Level 5 of Figure 5-12) steps through each configuraton within the group.

The step-by-step procedure for performing this algorithm is described as follows:

1) First, a file called PBDCL-MM (previous best desired channel loading mismatch) is initialized with the DCL3 data derived from Section 5.5.2.3.3 for each channel (DCL3 = 59.2 amperes for each channel).

2) Next, the switch configurations are set to their initial positions. The initial position for each load bus corresponds to Switch 1 being on, as illustrated by Table 5-7. This step also initializes a base three counter that is used to step through each of the possible combinations for the first group of five loads, as shown in Table 5-8. A base three counter is used since two of the load buses have three switches. The switch configuration is stored in a file called (SWT), as shown in Table 5-8.

3) After the switch positions are initialized, the channel connections are updated from the channel connection file in the load priority table (CHC-LPT), shown in Table 5-7. The updated channel connections are stored in a temporary file called (CH-CON), as shown in Table 5-8. Since most of the load buses do not have three switches, it is possible for the base three counter to obtain states that are not realizable. This condition is verified during this step when the channel connections are updated. If a nonrealizable state is detected, the counter is incremented to the next state and the channel connection update procedure is again performed. The base three counter operation is illustrated in Figure 5-15 for the first group of load buses in Table 5-8.

4) Once the channel connections have been updated for the first group of load buses, then the load current for each channel of the first group is summed and subtracted from the value in the PBDCL-MM file for each channel. (Initially this number is DCL3 as described in Step 1.) The results are the desired channel loading mismatches and these three values are stored in a file called DCL-MM (desired channel loading mismatch).

5) The DCL-MM numbers are then compared with each other to determine which one is the maximum. The maximum DCL-MM is stored in a temporary file called MAX-MMT. If the maximum DCL-MM is less than the previously recorded maximum DCL-MM, then the new maximum DCL-MM will be recorded in a permanent file called MAX-MM. (Since this is the first time through in this example, the value will automatically be recorded in the MAX-MM file.) If a smaller MAX-MM is obtained, then this configuration represents the best configuration that has been evaluated so far and the channel loading recorded in DCL-MM is transferred to a file called BDCL-MM (best desired channel loading for minimum mismatch) and the switch configuration is recorded in a file called BSWT (best switch configuration). The overall process

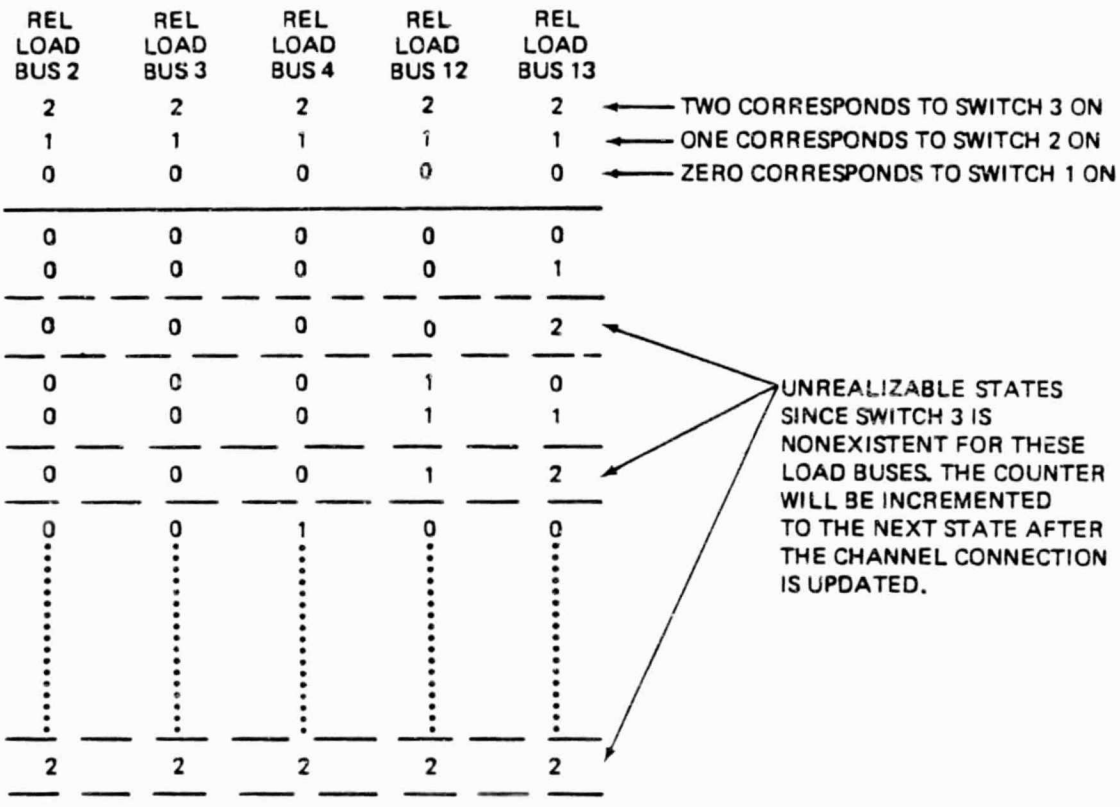


Figure 5-15. Illustration of Base 3 Counter Operation for Load Bus Assignments

is illustrated in Figure 5-16 for the first group loads in Table 5-8 with the initial switch positions.

6) The switch positions are now incremented by means of the base three counter, the channel connections are again updated, the realizable state is verified, and Steps 4 and 5 are repeated. This entire process is repeated until all the possible combinations for the first group load buses have been evaluated for the minimum mismatch.

7) After the first group loads have been completely evaluated, the best desired channel loading that can be obtained with the first group load buses is stored in BDCL-MM and the corresponding best first group load bus switch configuration is stored in BSWT. At this time, the PBDCL-MM file is updated with the BDCL-MM data, which forms the starting point for the second group load bus evaluation.

8) The entire process is now repeated for following group load buses until the best configuration is obtained.

5.5.2.3.5 Disconnect Load Buses for Overloaded Channels

For the case where the best configuration option results with one or more of the power channels overloaded, the software must identify and alleviate this condition. The disconnect load buses software will address this issue by surveying the overloaded channels to determine if there are disconnect controllable load buses (i.e., Priority 4 in the system load priority of the load priority table, SLP-LPT) which, if disconnected, would remove the overloaded condition. If no satisfactory solution is found with the normal priority settings, the mode of operation would be switched to the emergency condition. Then the analysis which is necessary to select the load buses for removal would be based on the emergency load priority information in the SLP-LPT. Once the overloaded condition is alleviated, it may be desirable to reevaluate the configuration options if the resulting maximum mismatch is larger than acceptable. This reevaluation decision will be made internal to this module. The software controlling these decisions has not been developed to date.

5.5.2.3.6 Setup Switching Sequence

As mentioned earlier in the load assignment methodology section, a switching sequence must be selected for the transfer of load buses from channel to channel such that overloading of the power channels is avoided. The setup switching sequence module will set up the sequence order that

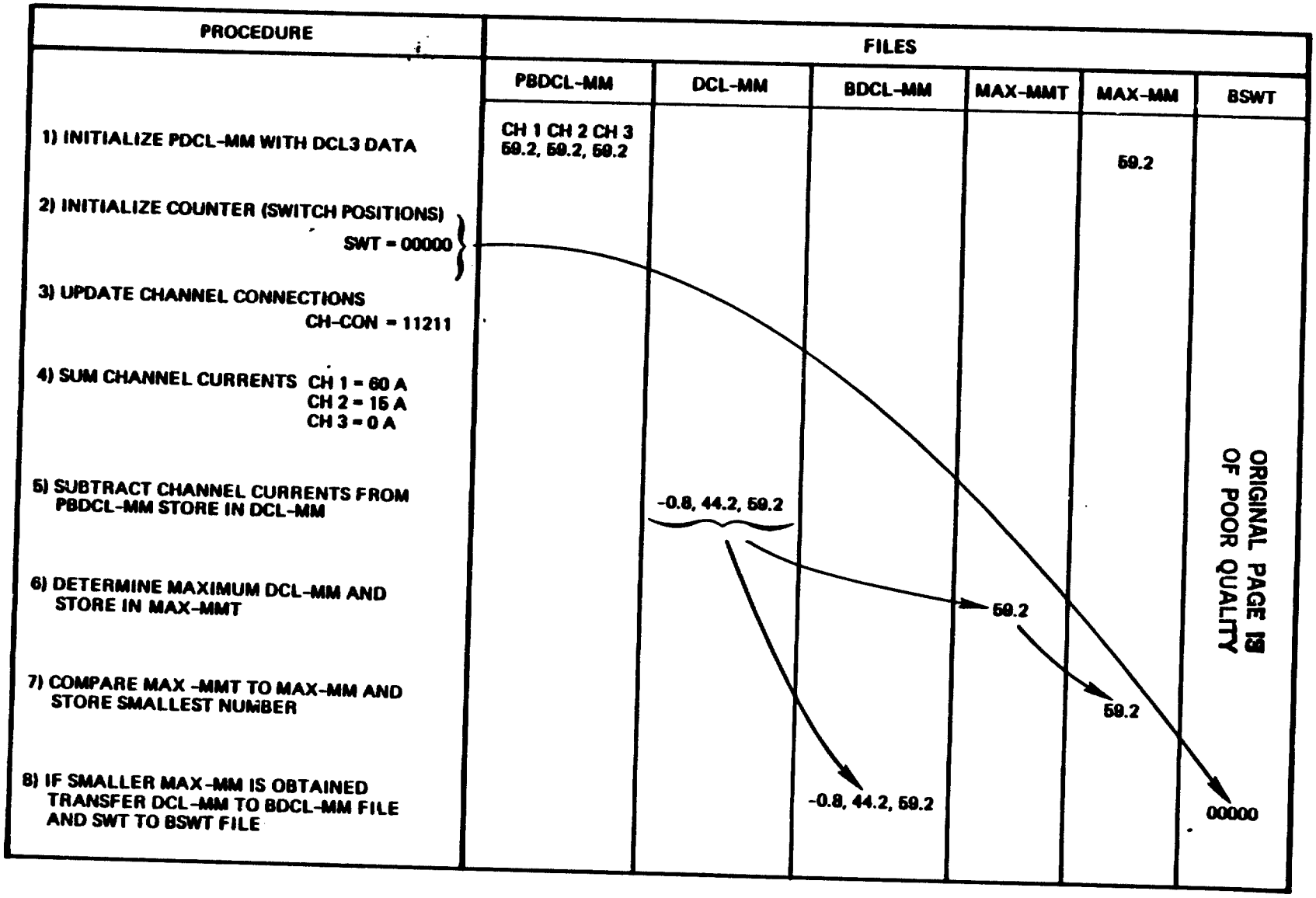


Figure 5-16. Illustration of Group Configuration Selection

commands be sent, order command verification, and reschedule load assignments if a faulty switch operation is reported. The presently programmed approach implements all of the above-mentioned processes except for the switching sequence analysis. The commands are processed one at a time in the order that the load buses were arranged for the load assignment group classification, starting with the load bus with the largest current first.

5.5.2.3.7 Output Load Bus Assignments Results (LA-RESULTS)

This process outputs the results of the load assignments analysis in two parts. The first part is the load bus data which is displayed in tabular form with the following headings:

a) Load Bus Number

b) Switch Number of "ON" Switch

c) Channel Connection

d) Load Bus Current.

The second part is displayed by hitting the "return" key on the keyboard/video display unit. This part is made up of the summary data for the channels, and is also displayed in tabular form with the following headings:

a) Channel Number

b) Desired Channel Loading

c) Resulting Channel Loading

d) Resulting Channel Mismatch.

The total number of configurations evaluated is also displayed along with this data.

The hierarchial structured design of this software module is illustrated in Figure 5-13. The design is mostly self-explanatory. with the process flowing from left to right. The first FORTH word "RESET-TCONF-EVAL" resets the TCONF-EVAL table to zeros. This table is used to total the channel loading as the load bus currents are displayed. The decision box in the LB-DATA module decides whether the load bus data has been updated by

the configuration selected or if it is the same as before the analysis was done. Each of the headings in the modules is the FORTH word defined to do the processing.

5.5.2.4 Load Bus Assignment Solution for Specific Example

The load bus assignment demonstration algorithm as described in Section 5.5.2.3 was run on the electrical power subsystem controller for the specific example of Figure 5-11 and Table 5-7. The results of this solution are shown in Table 5-9. The results show that the channels were balanced to within 2.9 percent of the desired load balance which would be well within the desired balancing accuracy of an actual system.

Table 5-9. Load Bus Assignment Solution for Specific Example

Load Bus	Switch On	Channel Connection	Load Current (amperes)
1-1	2	2	10
1-2	1	3	15
1-3	2	1	15
1-4	1	2	15
1-5	2	2	7.5
1-6	2	1	7.5
1-7	2	1	7.5
1-8	2	2	3.75
1-9	2	1	3.75
1-10	2	1	3.75
2-1	2	2	10
2-2	1	3	15
2-3	1	3	15
2-4	2	1	15
2-5	1	3	7.5
2-6	2	1	7.5
2-7	1	2	7.5
2-8	1	3	3.75
2-9	1	3	3.75
2-10	1	2	3.75

Channel Number	Desired Loading (amperes)	Resulting Loading (amperes)	Resulting Mismatch (amperes)
1	59.2	60	-0.8 (-1.3%)
2	59.2	57.5	1.7 (2.9%)
3	59.2	60	-0.8 (-1.3%)

A second example was constructed by changing the load current on load Buses 1-1, 1-2, and 1-10 to 12.9, 13.7, and 5.3 amperes, respectively. The load bus assignments algorithm balanced these loads to within ±0.3 percent, as illustrated in Table 5-10.

Table 5-10. Second Example of Load Bus Assignments Algorithm

Load Bus	Switch On	Channel Connection	Load Current (amperes)
1-1	3	1	12.9
1-2	2	2	13.7
1-3	2	1	15
1-4	1	2	15
1-5	1	3	7.5
1-6	1	3	7.5
1-7	2	1	7.5
1-8	1	3	3.75
1-9	2	1	3.75
1-10	1	2	5.3
2-1	3	1	10
2-2	1	3	15
2-3	1	3	15
2-4	1	2	15
2-5	1	3	7.5
2-6	2	1	7.5
2-7	1	2	7.5
2-8	1	3	3.75
2-9	2	1	3.75
2-10	1	2	3.75

Channel Number	Desired Loading (amperes)	Resulting Loading (amperes)	Resulting Mismatch (amperes)
1	60.2	60.4	-0.2 (-0.3%)
2	60.2	60.2	-0- (0%)
3	60.2	60	+0.2 (+0.3%)

These two examples illustrate that the mathematical procedure for balancing the loads is correct. The algorithm was capable of balancing these loads very accurately. However, in general, the balancing accuracy will depend on the number and magnitudes of the loads which are available for switching without system implied constraints.

5.5.3 Power Subsystem State-of-Health

The power subsystem state-of-health algorithm maintains the overall status of the power system by performing the following functions:

a) Assembles and updates the summary data which determines the battery and solar array states of health.

b) Interprets fault messages from the power subsystem, makes the appropriate adjustments in the tables, and takes the necessary control action.

c) Interprets information requests from the spacecraft controller, assembles the requested data, and sends it back to the spacecraft controller.

d) Interprets commands from the spacecraft controller and issues them to the appropriate power subsystem controller.

e) Assembles and updates load priority tables based on startup, emergency and new load conditions, and telemetry and status data.

f) Provides command control for setting and verifying the proper circuit breaker current limits.

The summary data for the battery state of health consists of the battery capacities, cell failures, and charge/discharge parameter limits. For the solar array state of health, the output power capabilities, string failures, and degradation status are included in this summary. The load priority tables effectively serve as the electrical distribution system state-of-health table. The structural design of the load priority tables is shown in Figure 5-17. The following load priority table files have been set up for the load bus assignment demonstration algorithm:

a) System Load Priorities for Load Priority Table (SLP-LPT)

This file stores the priority of each load bus for normal and emergency operation. One byte is required for each priority; therefore 2 bytes per load are used.

Priority definitions used:

0 - no load

1 - nonswitchable load

2 - semiswitchable load

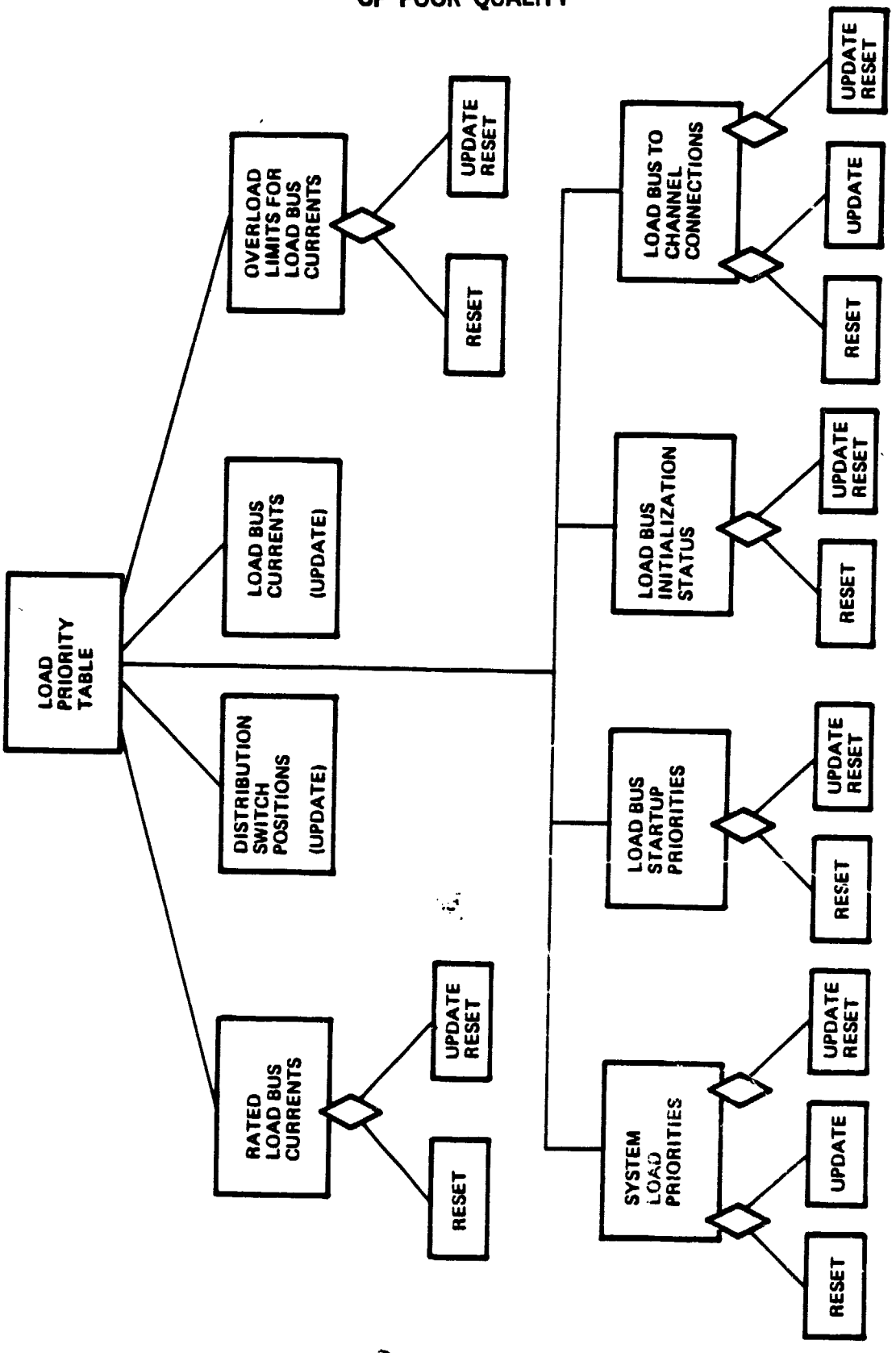


Figure 5-17. Load Priority Table Structural Design

C-2

3 - switchable load

4 - disconnect control.

This file must be set on start-up and is alterable via spacecraft controller interfaces.

b) Switch Positions for Load Priority Table (SW-LPT)

This file stores the load bus switch status for each of the load buses. One byte is used for each load bus with each bit corresponding to a designated bus switch. (Example; the first bit is for the first switch, the second bit is for the second switch, etc.) A one is stored for the switch ON and a zero is stored for the switch OFF in the respective bit.

This file must be set by data transfers requested by the EPSC from the LCCs.

c) Channel Connection for Load Priority Table (CHC-LPT)

This file stores sequentially the channel connection for each switch of each load bus. Each switch has 1 byte allocated and each load bus has memory space allocated for three switches. The number of the channel which the switch connects the load bus to when it is turned on is stored in this file. A zero is stored to signify no switch or no channel connection. This is used in the case where less than three switches actually exist for a given load. This file must be set on start-up and is alterable via spacecraft controller interfaces.

d) Load Bus Current for Load Priority Table (LC-LPT)

This file stores the actual current values for each of the load buses. Each load current is stored sequentially with one data location cell required for each load current.

5.5.4 EPS Management Processing Requirements

The EPS management algorithms were analyzed to determine the computer processing requirements in terms of the number of instructions/second that are needed to perform the algorithms. This analysis required determining the number of times per second that the algorithm must be repeated and the total number of programming instructions required to define the algorithm. This analysis is summarized in Table 5-11.

The repeat time for the energy planning and allocation and load bus assignments algorithms was based on the 90-minute spacecraft orbit with 36-minute eclipse and 54-minute daylight periods. A repeat time of every 5 minutes was selected based on an engineering judgment that the energy

Table 5-11. Electrical Power Subsystem Management Processing Requirements

Functions	Instructions per Function	Repetition Interval (sec)	Controller Instruction Rate (instr/sec)	System* Instruction Rate (instr/sec)
Spacecraft and Executive	8,700	1	8,700	17,400
Power Subsystem State of Health	103,000	60	1,717	3,433
Energy Planning and Allocation	8,000	300	27	53
Load Assignments	5,400,000	300	18,000	36,000
			28,444	56,886

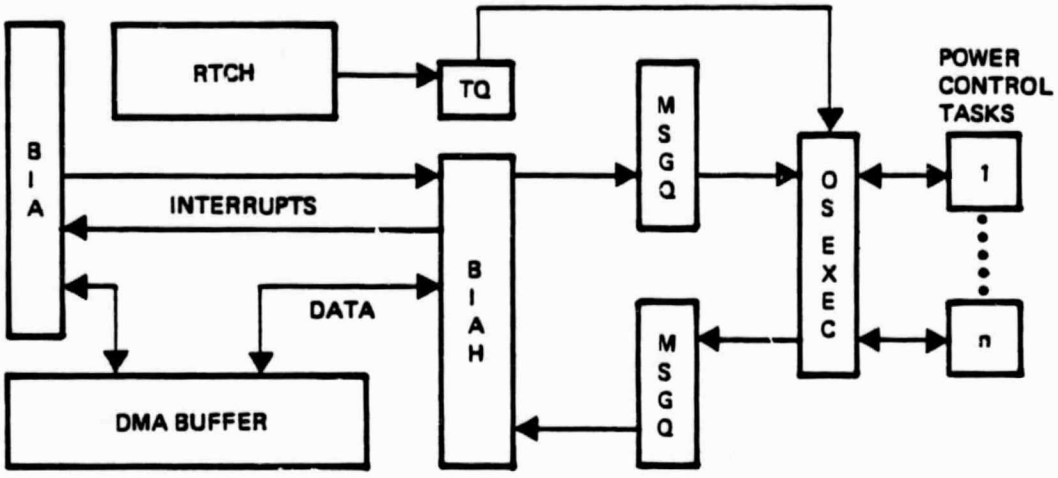
*Two subsystem controllers operating simultaneously; one redundant

balance of the electrical power subsystem should be calculated six to seven times during each period. This judgment was made based on the fact that the electrical power subsystem is a utility-type system in which the load profiles are unknown and are constantly changing. Therefore corrective action may be required at any time during the orbit to maintain specified bus voltage limits. The power subsystem state-of-health algorithm was arbitrarily selected to operate once per minute. The executive algorithm which schedules the overall tasks for the EPS controller was selected to operate once per second.

The number of instructions required to perform an algorithm were calculated based on past experience in programming similar algorithms for spacecraft applications.

5.6 EXECUTIVE SOFTWARE

Executive software was developed for the power management system controllers based upon the development system inherent in the selected operating system (FORTH). An interface diagram of the software components is shown in Figure 5-18. A description of the software components is presented in the following sections.



BIA	BUS INTERFACE ADAPTER
BIAH	BUS INTERFACE ADAPTER HANDLER
DMA BUFFER	DATA STORAGE BETWEEN BIA AND BIAH
INTERRUPTS	INTERRUPTS BETWEEN BIA AND BIAH
MSGQ	DATA QUEUES BETWEEN BIAH AND OS EXECUTIVE
OS EXEC	OPERATING SYSTEM EXECUTIVE LOOP
RTCH	REAL-TIME CLOCK HANDLER
TQ	TIMER QUEUE SET BY POWER CONTROL TASK PROCESSED BY OS EXECUTIVE

Figure 5-18. Software System Interface Diagram

5.6.1 BIA Handler (BIAH)

The BIA handler provides the communication link among the power controllers. When the BIA receives data from the data bus, it stores the data received in its local buffer. The received data is then verified and the data transferred to the DMA buffer. The BIA then interrupts the BIAH and informs the BIAH that data is available for processing. Upon receiving the interrupt the BIAH checks the receive/error status; if no error, the received data is then moved to the system input queue (MSGQ in Figure 5-18) for further processing by the operating system (OS) executive. If an error is detected, the BIAH takes appropriate error recovery procedures.

5.6.2 Real-Time Clock Interrupt Handler

The Real-Time Clock Handler (RTCH) increments the counter within the timer queue (TQ, Figure 5-18) upon receipt of a hardware clock interrupt (for example, every 10 milliseconds). The timer queue is a file of task identification words and timer counter values associated with the power control tasks. The operating system reads the timer queue counter and the task file to determine the next task/routine to be performed.

5.6.3 Operating System Executive

The operating system executive operates in a round-robin manner. While not serving the RTCH or BIA interrupts, the operating system executive runs through the executive loop looking for work to dispatch. If any one of the functions is dispatched, the program returns to the start of the executive loop. If all functions are checked and nothing is dispatched the CPU idle counter is incremented at the end of the loop. Figure 5-19 is a flow diagram of the executive loop.

Each executive function is described in the following subsections.

a) Process BIAH to Message Queue (MSGQ)

Data received from the BIA are queued by the BIAH into the operating system message queue. The operating system processes the data in the queue one at a time by activating the corresponding power control task (type) that is identified in the queue entry. The following is a definition of the entry:

MESSAGE OR COMMAND TYPE - MESSAGE TEXT

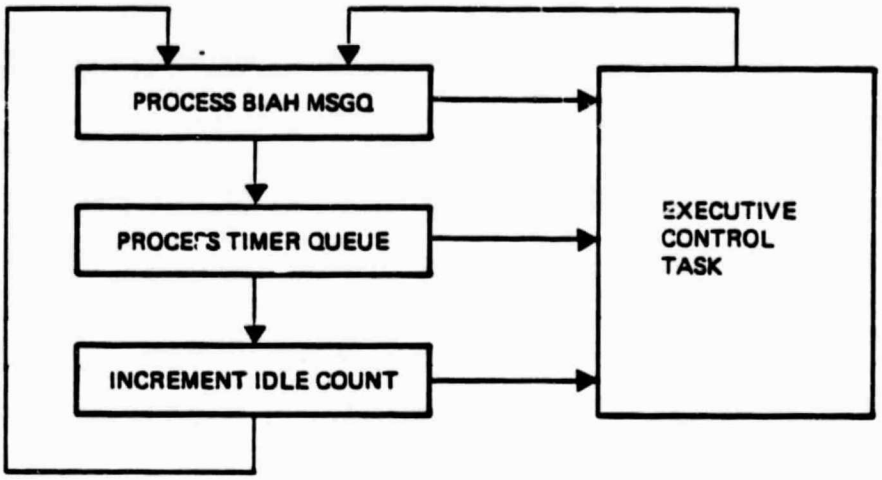


Figure 5-19. Flow Diagram of Executive Loop

If any task is dispatched, return is to the start of the executive loop. If there is no entry in the queue, the next function in the loop is served.

b) Process Timer Queue (TQ)

The timer queue entries are examined in sequence until a time-out "one" is found. The task identification for this time-out entry is dispatched and the return is to the start of the executive loop. If there is no time-out event in the queue, the control is passed to the increment idle count process (Figure 5-19).

c) Increment Idle Count Process

If the executive loop reaches this function then no other function was ready to run and an idle state is indicated. The executive increments the idle counter. This counter can be used to quantify the activity within a power controller.

6. POWER MANAGEMENT SUBSYSTEM

A decentralized data processing approach was selected for the power management system based on the electrical power subsystem (EPS) algorithms that are to be performed and the control hierarchy trade studies and analyses described in Section 8.2. The components of the power management subsystem (PMS) and their relationship to the electrical power subsystem (EPS) and the autonomously managed power system (AMPS) are described in the following sections.

6.1 FUNCTIONAL DESCRIPTION OF THE ELECTRICAL POWER SUBSYSTEM

The major components and interfaces of an EPS with automated power management capability are shown in Figure 6-1. As defined for this project, the EPS is made up of energy storage devices, power generation devices, the AMPS, and thermal processing equipment. The energy storage devices are nickel-hydrogen batteries and the power generation device consists of solar array panels. The thermal processing equipment provides the means to control and maintain the spacecraft within specified temperature limits. The thermal processing system is a pumped fluid system. The AMPS is described below.

6.2 FUNCTIONAL DESCRIPTION OF THE AUTONOMOUSLY MANAGED POWER SYSTEM

The Autonomously Managed Power System (AMPS), Figure 6-1, consists of the Power Management Subsystem (PMS) and the devices that distribute, process, and manage the power delivered from the power generation and energy storage devices. The building blocks for the power processing, conditioning, protection, and distribution functions are the energy storage electronics, power generation electronics, primary power distribution, power conditioning, and secondary power distribution modules. Included in each of these modules are the input/output (I/O) circuits that are part of the PMS. The energy storage electronics are responsible for the processing of the power flow to and from the energy storage devices so that proper charging, discharging, and maintenance can be accomplished.

Battery reconditioning and cell failure detection circuitry are also included in the energy storage electronics. The function of the power generation electronics is to process the power from the solar array panels.

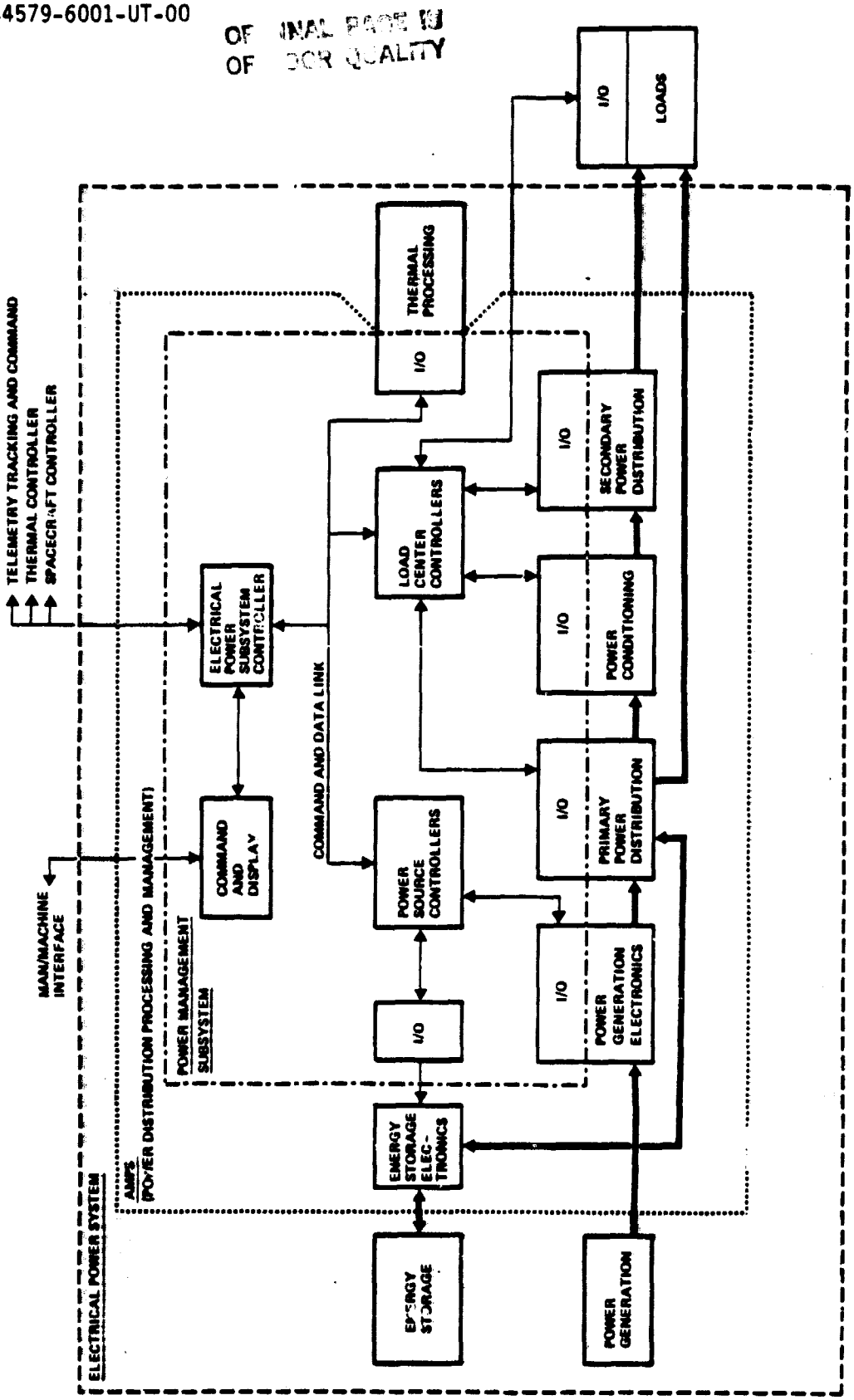


Figure 6-1. Autonomously Managed Electrical Power System

Included in the power generation electronics is the electrical isolation of each solar array section and the circuitry for limiting or regulating the solar array voltage. The primary power distribution equipment provides the tie points of the solar array and the batteries to the main bus. Included in the primary power distribution equipment are power switches, protection devices, and power processing equipment required for the main bus operation. Power output from the primary power distribution equipment is through load buses to either the power conditioning units or the loads. The power conditioning units convert the main bus voltage to the feeder voltages that are required by the loads. These units in general are high efficiency switching regulators, i.e., converters or inverters that provide dc-to-dc or dc-to-ac voltage conversion, respectively. Secondary power distribution hardware provides the distribution, switching, and protection for the power conditioning output voltages. The PMS is described below.

6.3 FUNCTIONAL DESCRIPTION OF THE POWER MANAGEMENT SYSTEM

The PMS monitors and controls the complete electrical power system from generation to load so that the monitoring, processing and decision, control, and recording functions can be efficiently and predictably performed. The PMS consists of I/O circuitry, power source controllers, load center controllers, electrical power subsystem controller, command and display interfacing hardware, and the data bus between the controllers.

The PMS is a decentralized processing system from the standpoint that the power source controllers, load center controllers, and I/O circuitry are distributed throughout the spacecraft at various functional centers. An example of the distributed processing concept for an autonomously managed EPS is shown in Figure 6-2. In this example, the power system controller and its redundant backup controller are located in the spacecraft control center where power system information is displayed for on-board personnel and is also available for further processing to the ground through the telemetry, tracking, and command (TT&C) subsystem. Power system commands are also received through the TT&C subsystem and are routed to the PMS controllers via the data bus. Power source controllers are located at power generation centers where solar array power is integrated with energy storage device power to form the main power buses for the spacecraft. The main buses are then routed to the various spacecraft load

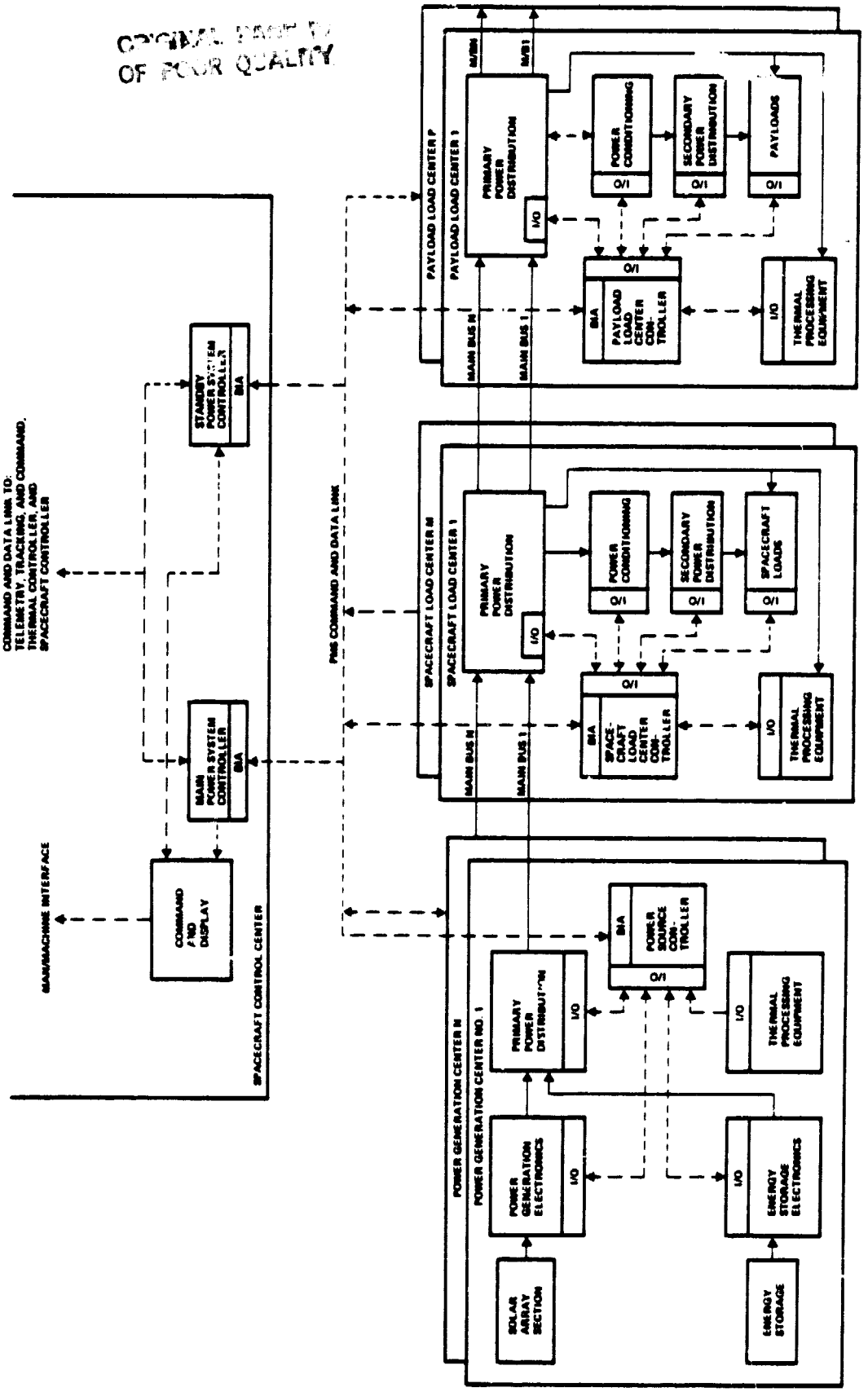


Figure 6-2. Example of Distributed Processing for an Autonomously Managed Electrical Power System

centers where the load center controllers are located. In the example, load centers for essential spacecraft loads have been distinguished from payload load centers. The payload load center controllers provide data processing for the payloads as well as the power system management. The thermal processing equipment is also distributed throughout the functional centers, and thermal data is processed by the respective distributed load center controllers.

6.3.1 I/O Circuitry

The I/O circuitry consists of sensors and signal processing circuitry. Monitoring functions are implemented by the use of sensors. Sensors measure various system parameters and generally transform these signals into analogs that are usable by the electronic signal processing circuitry. The outputs from the sensors are analog signals that are proportional to the input parameters or the status of devices such as relay contacts. The analog sensors are transducers that monitor voltage, current, pressure, or temperature.

The signal processing circuitry gathers data from the sensors and processes this data so that it can be used by the power source or load center controllers. The signal processing circuitry also processes commands from the controllers and routes them to the appropriate actuator. Specific functions of the signal processing circuitry are: multiplexing of sensor channels, analog-to-digital conversion, digital-to-analog conversion, encoding and decoding, and logic operations.

6.3.2 Controllers

Three types of microprocessor-based controllers are defined in the selected power management subsystem concept: electrical power subsystem controller (Figure 6-3), power source controller (Figure 6-4), and load center controller (Figure 6-5). The primary function of each controller is to generate logical control decisions for the operation of the EPS and to manipulate sensor and other input data. To accomplish this, logical processing and data storage functions are required. The microprocessor and memory elements of each controller implement these functions. These elements also provide for the implementation of the algorithms that govern the control operations. The remaining major elements in each controller are

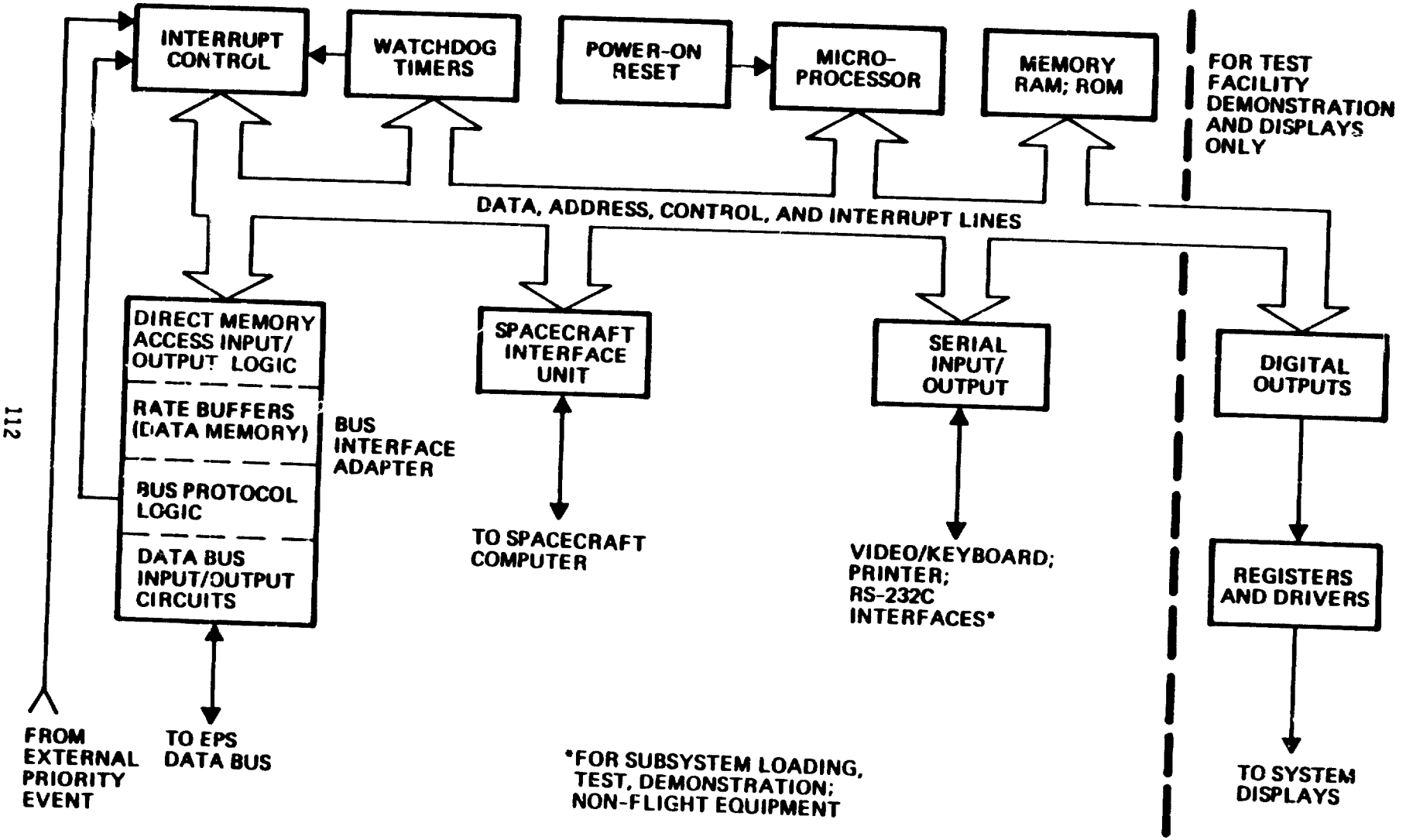


Figure 6-3. Electrical Power Subsystem Controller Architecture

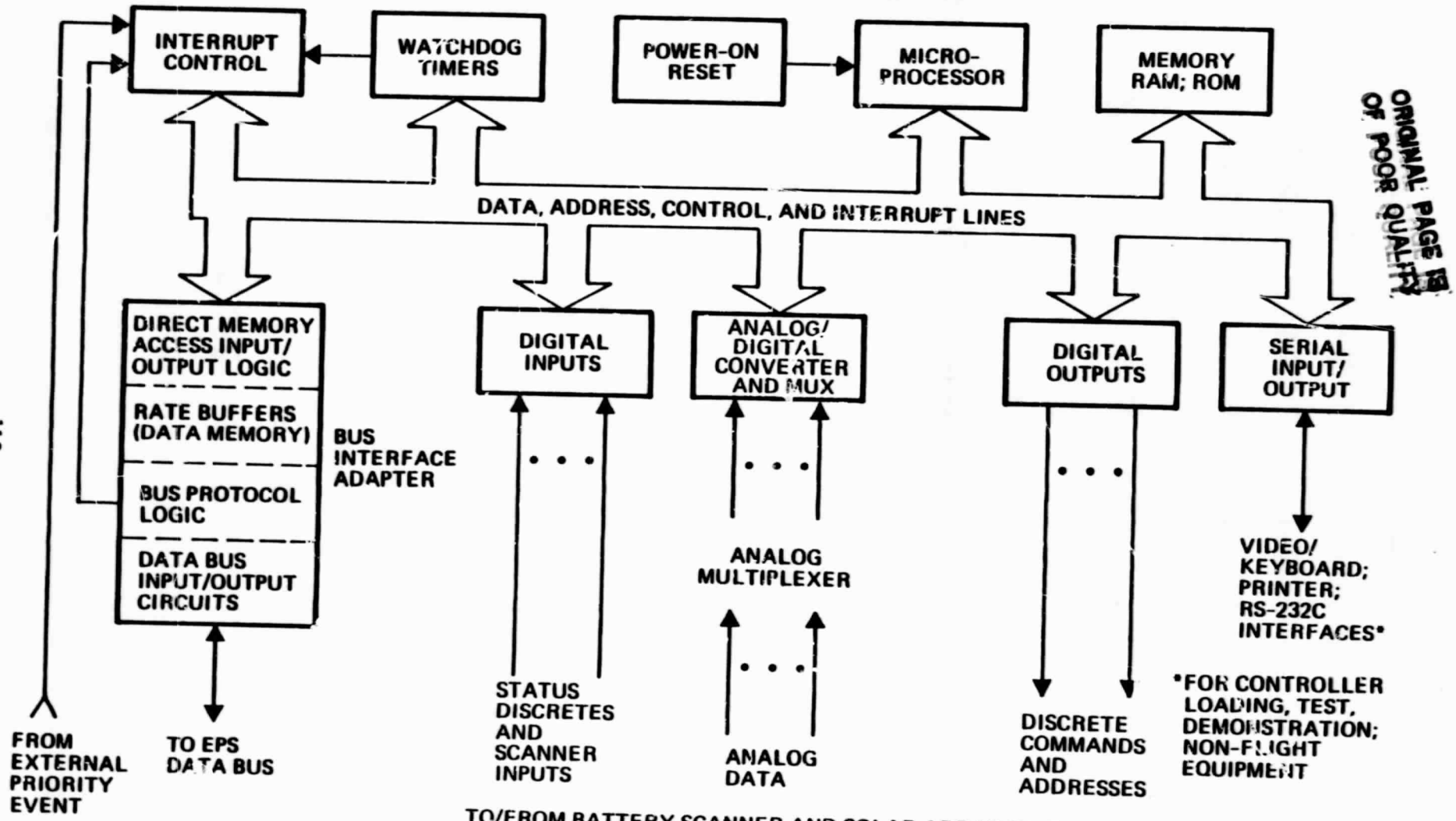


Figure 6-4. Power Source Controller Architecture

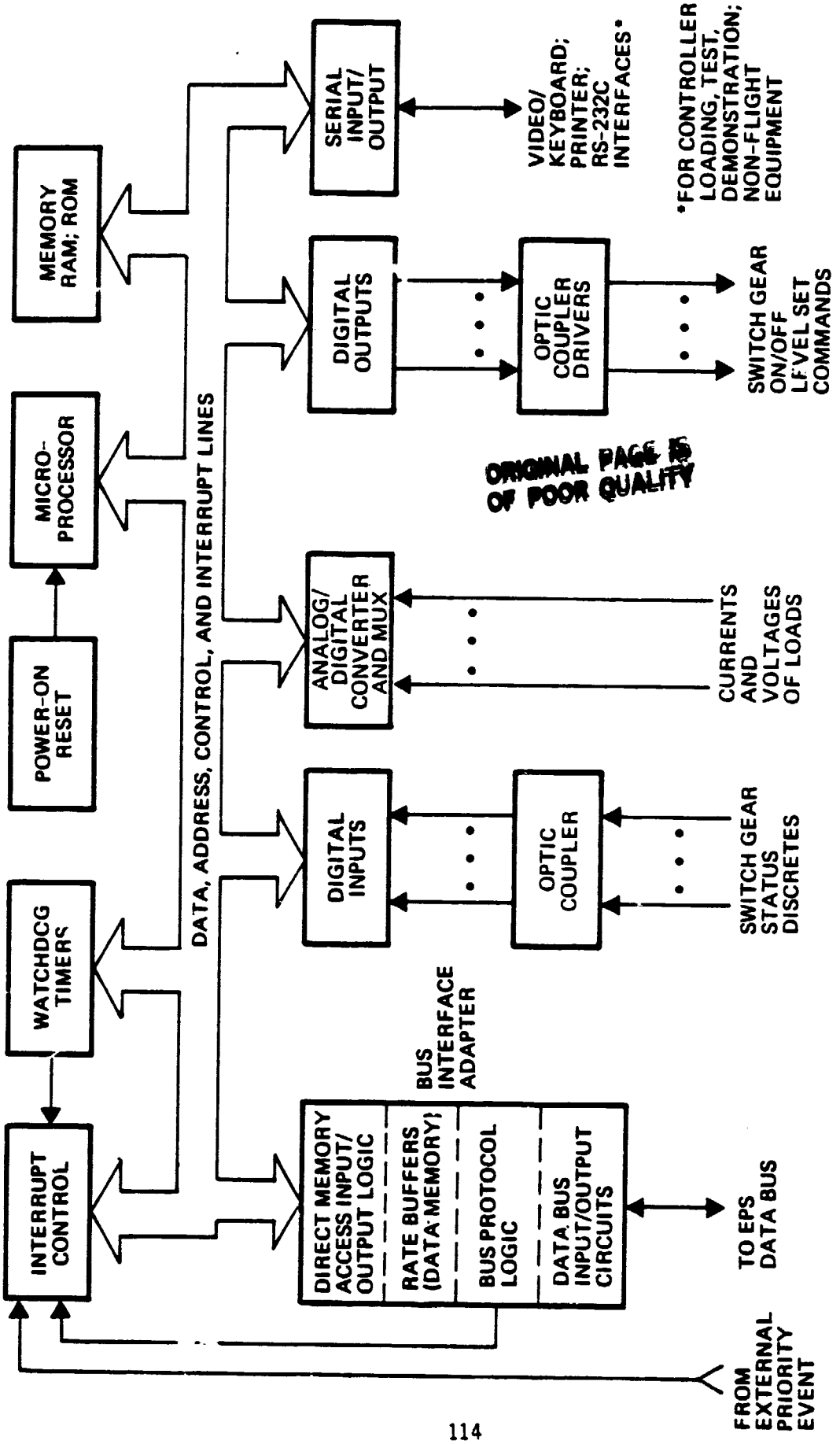


Figure 6-5. Load Center Controller Architecture

input/output oriented and dependent upon the execution and implementation requirements of the control functions assigned to each controller.

Each controller may communicate with the other controllers over a common data bus. A bus interface adapter (BIA) in each controller interfaces this communication path as a two-way device. It formats the output data into an appropriate message structure (rate, format, time slot) for communication to other controllers; it also recognizes a proper input message structure and decodes this message into data/command format. The bus interface adapter consists of four sections: bus input/output circuits, bus protocol logic, data buffers, and direct memory access input/output logic. The bus input/output circuits include line drivers and receivers, and encoders and decoders for the serial data stream of the data bus. The bus protocol logic performs serial-to-parallel conversion and other low level functions of the high level data link; for example, special character decoding (message flag sequence and abort flag), zero bit stuffing and deletion, error checking, and interrupt generation. The data buffers provide double buffering capability for the maximum length data frame. The direct memory access input/output logic provides high speed data transfers between the bus interface adapter data buffers and the microprocessor memory.

Additional elements are included in each controller to provide peripheral features such as power-on reset, timing checks to escape from a nondefinitive or excessively lengthy processing routine, and interrupt procedures to prioritize processing routines. A digital input/ output port is provided on each controller for a serial data interface to provide a human interface to the system. Peripheral devices, such as printers and video/keyboard terminals, are connected to the system only during development, testing, and demonstration.

The Electrical Power Subsystem Controller (Figure 6-3) implements the management functions in the electrical power subsystem: for example, energy planning and allocation, load assignments, and command and data interfacing to other spacecraft subsystems. These functions require communications

with other EPS controllers via the PMS data bus, and with other spacecraft subsystem controllers and the spacecraft computer via the spacecraft data bus.

Included in the EPS controller is a digital output port to drive a subsystem status display for human monitoring of the EPS status and operation during system demonstration. Test and verification of flight hardware is through the video/keyboard terminal interface.

The Power Source Controller (Figure 6-4) monitors the solar array and battery operation and controls the solar array output to provide the desired battery charging and load currents. Digital output ports are required to control switches to adjust the solar array outputs and to select the desired battery operating mode: charge/discharge, recondition discharge, discharge/disconnect shutdown for servicing. Digital and analog input ports are required for the status and sensor data from the solar array and battery. The analog input ports include an analog-to-digital converter for the battery cell voltage, temperature, pressure, and current sensor data. This digitized data and the discrete status of the solar array switches are entered into the controller memory for future reference in the command decision processing.

The Load Center Controller (Figure 6-5) commands the load center switch-gear and monitors the results (both switchgear response and load parameters). Discrete commands control switchgear on/off operation, and parallel digital commands set the overload shutdown parameters of the switchgear. The switch gear status (on, off, tripped) and the actual load-bus current and voltage (analogs) are monitored. The data is thereby available in the controller for determination of switch gear operation and (if required) definition of any corrective action required (new commands).

6.3.3 Command and Display

The command and display circuitry provides the interface with the personnel for the monitoring and control of the power system during ground test or for application with a manned spacecraft. Power system conditions are displayed by means of status indicators and information is provided through readout devices such as video displays. Commands are issued through input terminals such as keyboards.

6.4 POWER MANAGEMENT SYSTEM DATA BUS

The PMS data bus provides for bidirectional information transfer between the PMS controllers. The data bus uses a global architecture in that any single controller can communicate directly with any other controller on the data bus. The selection of the global architecture is based on the trade studies and analyses presented in Section 8.3. The global architecture works in conjunction with a distributed-control, time-sequential, data bus contention resolution scheme and the International Standards Organization (ISO) high-level data link control (HDLC) data bus protocol to provide the overall data communications design. Selection of the data bus contention resolution and the HDLC protocol is based on the trade studies and analysis presented in Sections 8.5 and 8.4, respectively.

7. HARDWARE DEVELOPMENT

Hardware development for this phase of the program consisted of the development of two controllers: an electrical power subsystem (EPS) controller and a load center controller. Each controller consists of a microprocessor, memory, input/output (I/O) circuitry, and a bus interface adapter (BIA). The microprocessor that was selected for the controllers is a Texas Instruments 9900 unit with 64K bytes of memory capacity. (See Section 8.6 for microprocessor selection tradeoffs.) Up to 20K of the memory can be ROM with the remaining portion in RAM. The microprocessor, memory, and I/O circuitry were assembled from commercially available Texas Instruments printed circuit cards. The cards are assembled as a unit in a Texas Instruments TM 990/530 sixteen-slot card chassis. The BIA is being developed on a TRW internal research and development program (IRAD). The BIA will be fabricated on a universal prototype board, as shown in Figure 7-1, which also fits into the card chassis. A picture of the overall system is shown in Figure 7-2.

7.1 ELECTRICAL POWER SUBSYSTEM CONTROLLER DEVELOPMENT

A block diagram of the EPS controller is shown in Figure 7-3. The controller consists of a BIA card (module), microcomputer card, and memory card. In addition, a floppy disk controller card is used as test equipment for development, test, and demonstration. The floppy disk was used to store the FORTH compiler and the power management system algorithms during this development phase of the project. A picture of the EPS controller is shown in Figure 7-4. Pictures of the microcomputer card and the memory card are shown in Figures 7-5 and 7-6, respectively. An interconnect diagram is shown in Figure 7-7. A detailed description of the printed circuit card can be found in Appendix B.

7.2 LOAD CENTER CONTROLLER DEVELOPMENT

A block diagram of the load center controller is shown in Figure 7-8. The controller consists of a BIA, microcomputer, memory, digital I/O, analog input, and DC I/O cards. A floppy disk controller card is also included in the load center controller as test equipment for development, test, and demonstration. The load center controller is shown in Figure 7-9.

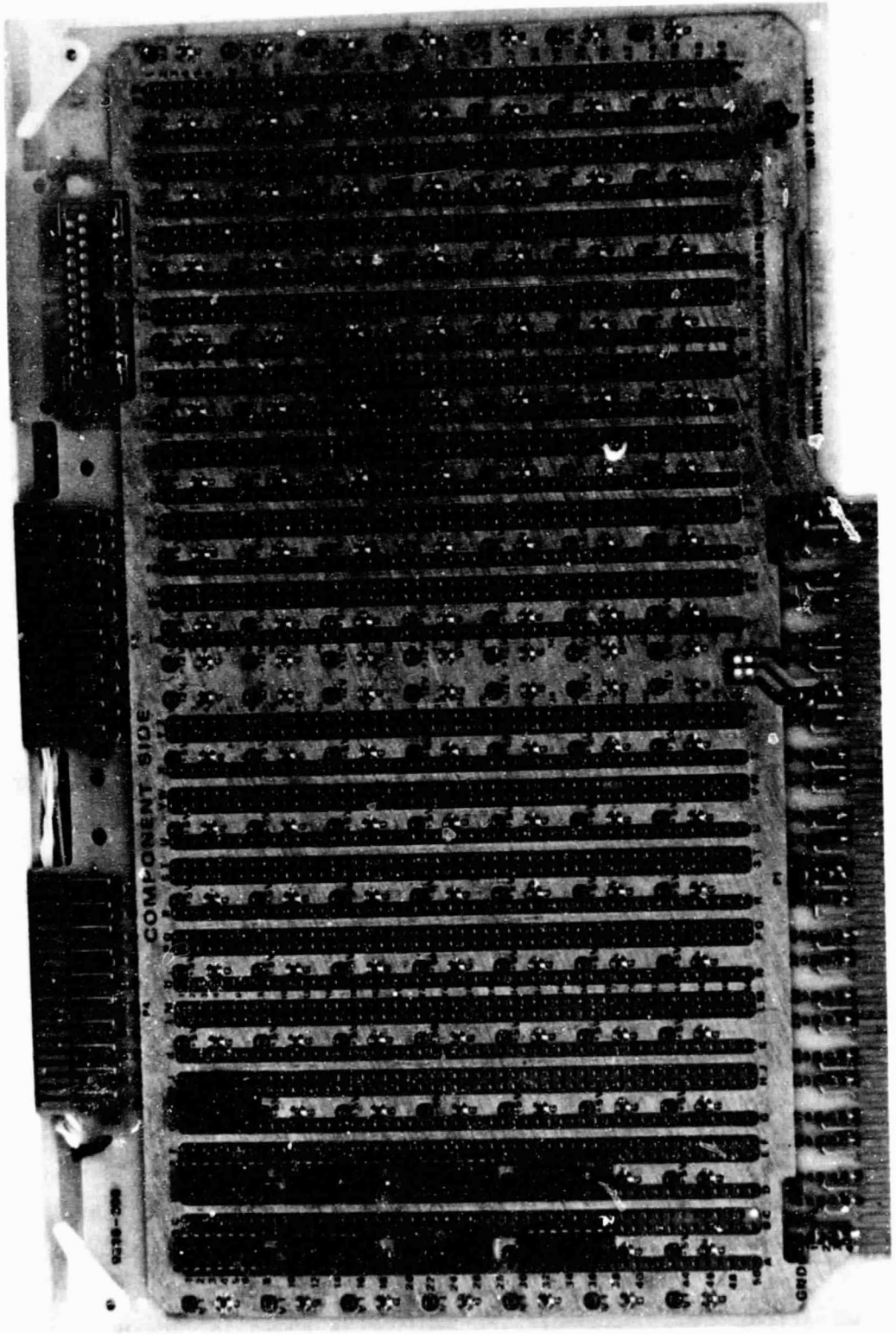


Figure 7-1. Universal Prototype Board (7.5 x 11 inch card)

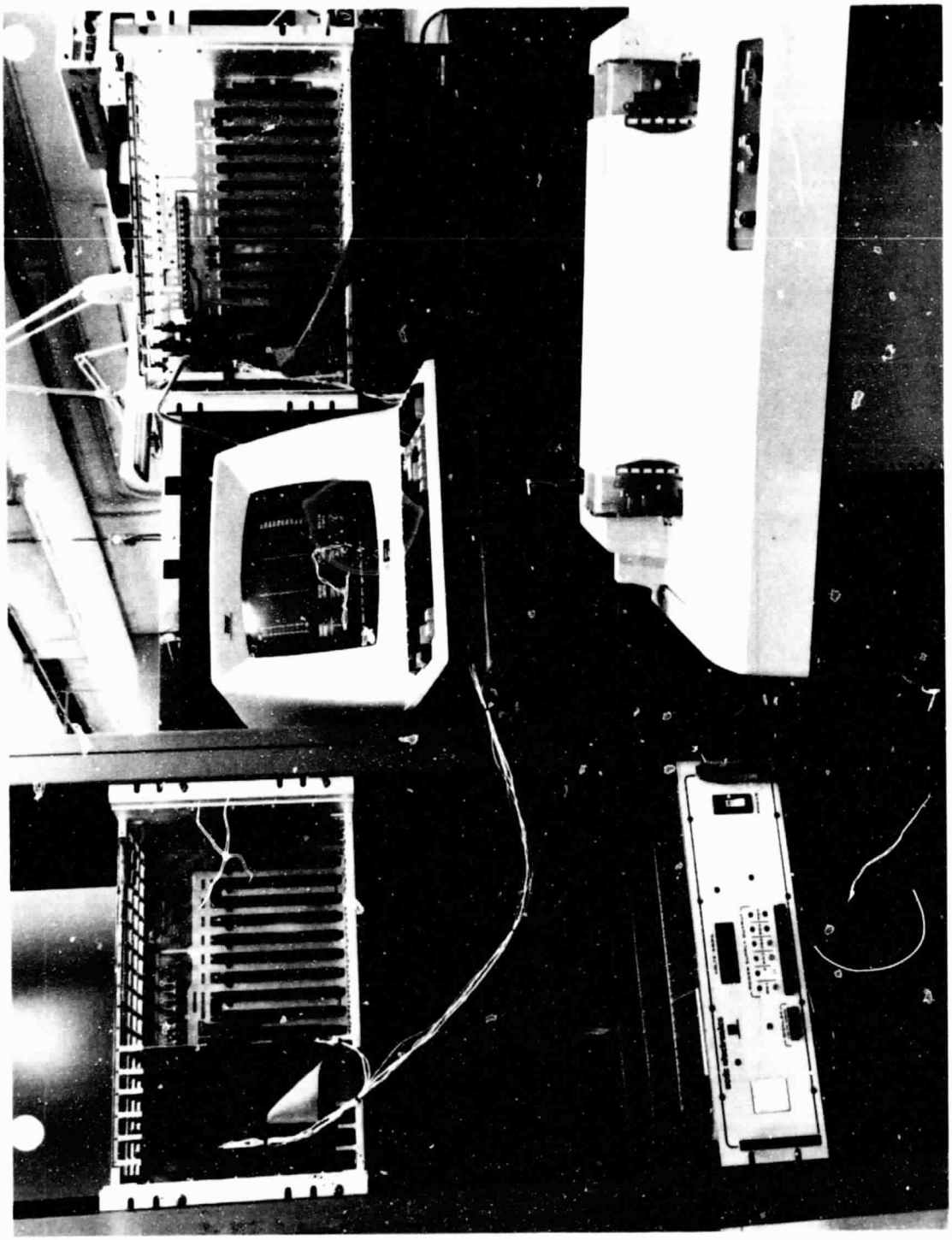


Figure 7-2. Power Management System Controllers

NOTES:

1. IN ADDITION TO THE TI 9900 MICROPROCESSOR, MEMORY, AND TWO RS232C I/O PORTS, THIS MODULE CONTAINS CIRCUITRY TO IMPLEMENT THE REAL-TIME INTERRUPT (RTI) CLOCK GENERATOR, TWO WATCH-DOG TIMERS, 16 EXTERNAL INTERRUPT INPUTS, AND A MANUALLY OPERATED RESET SWITCH.
2. SHADED BLOCK ON TEST EQUIPMENT FOR DEVELOPMENT, TEST, AND DEMONSTRATION ONLY.

Figure 7-3. Electrical Power Subsystem Controller Block Diagram

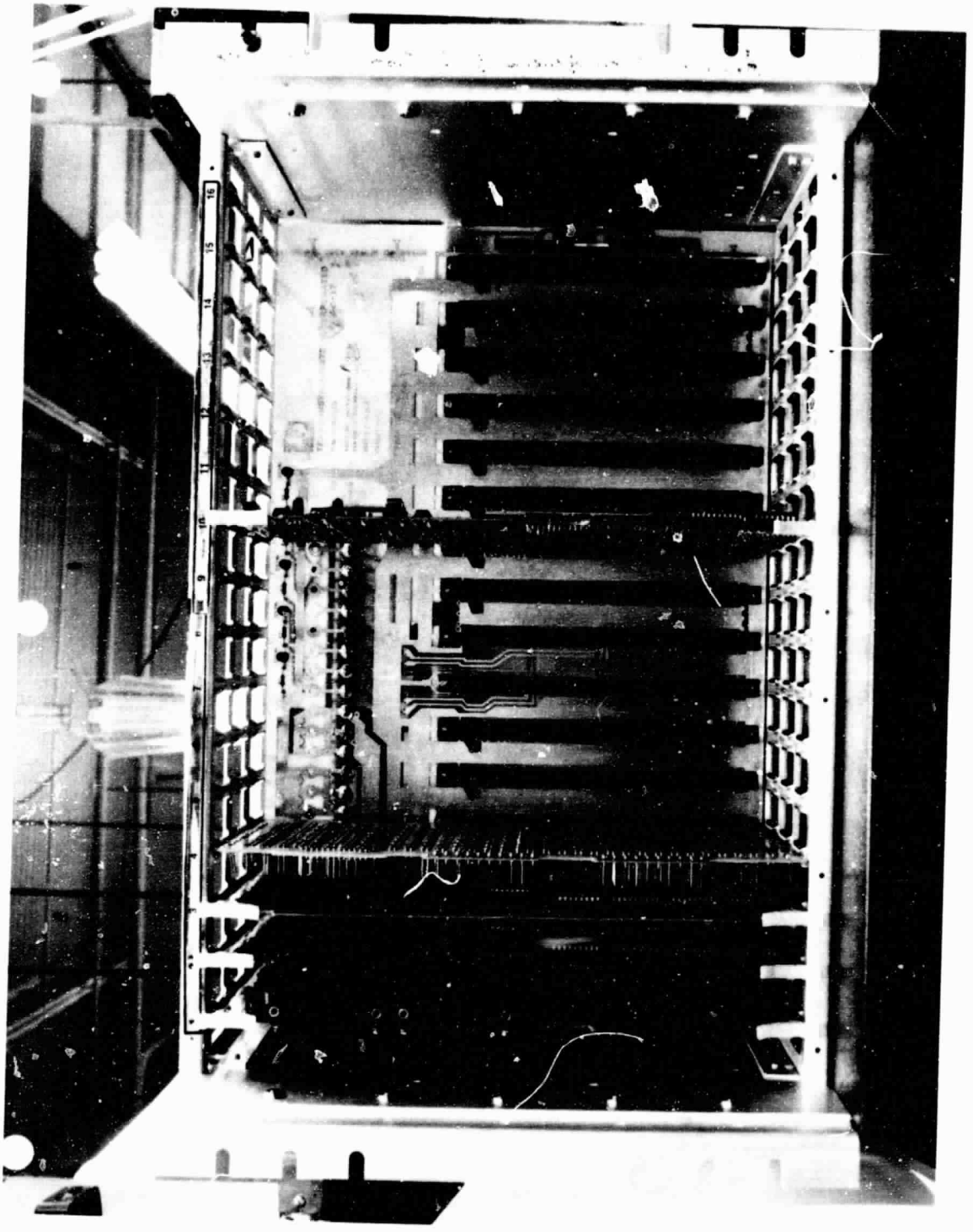


Figure 7-4. Electrical Power Subsystem Controller (19 inch rack mounting)

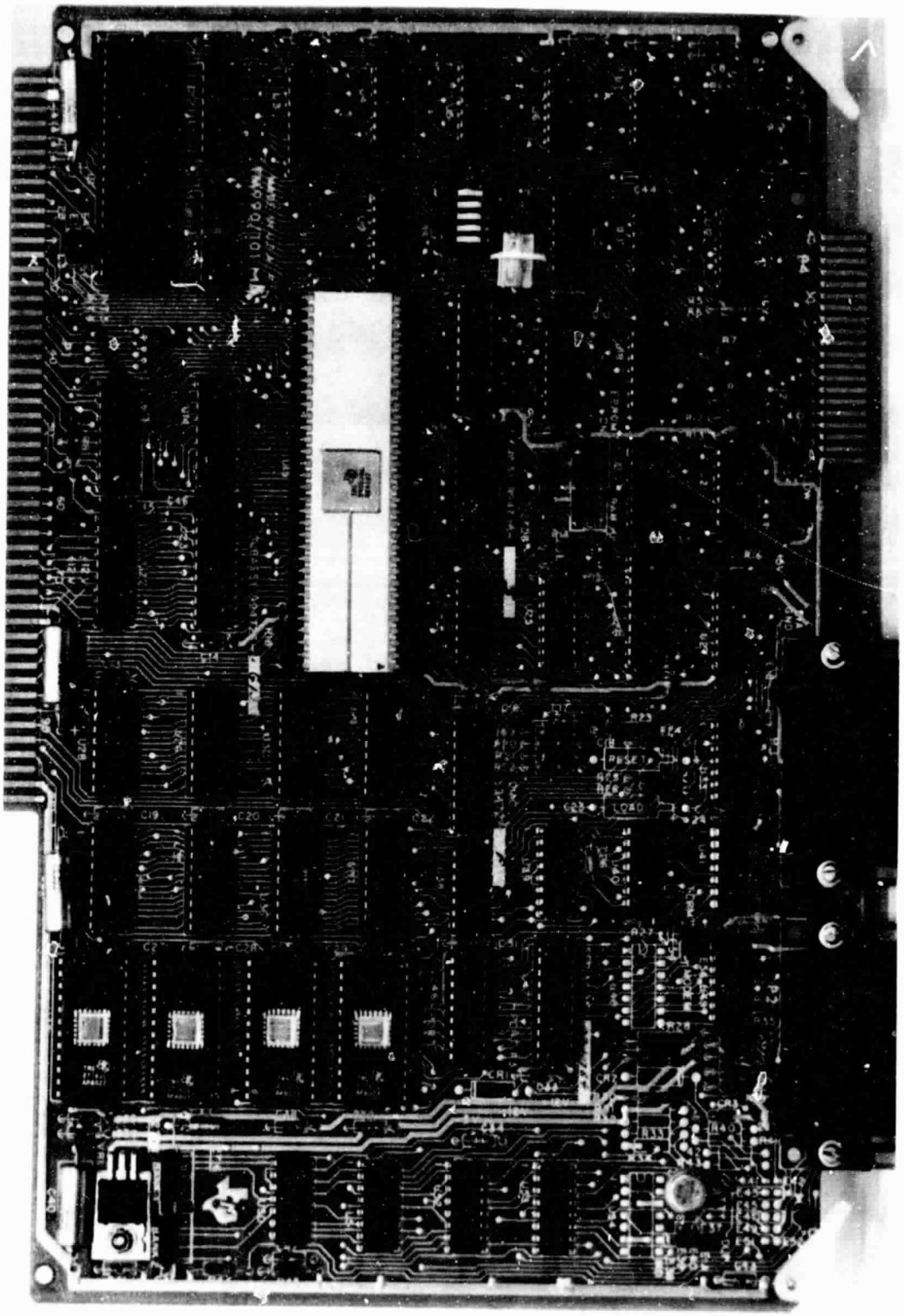


Figure 7-5. Microcomputer Card (7.5 x 11 inch card)

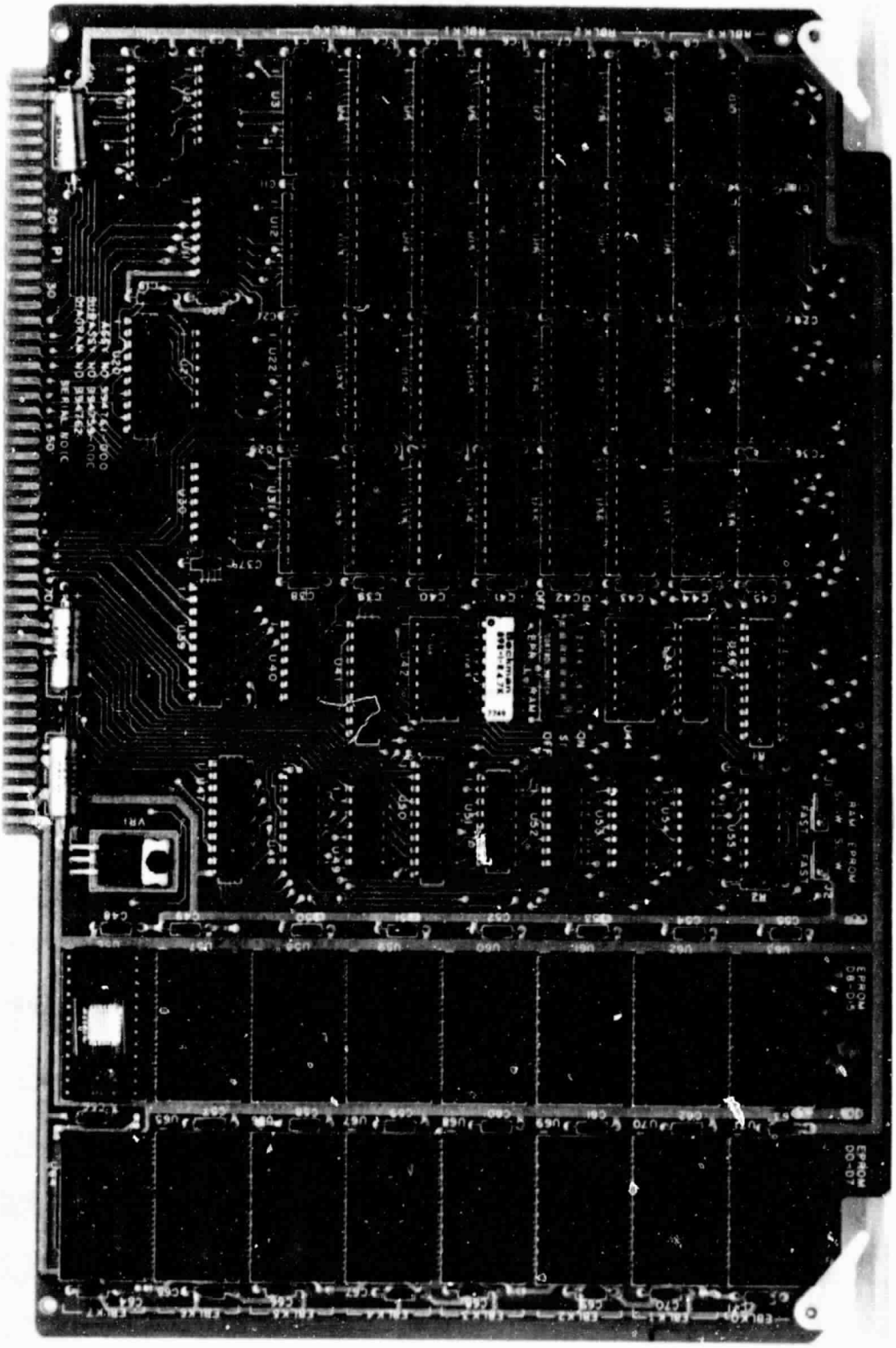


Figure 7-6. Memory Card (7.5 x 11 inch card)

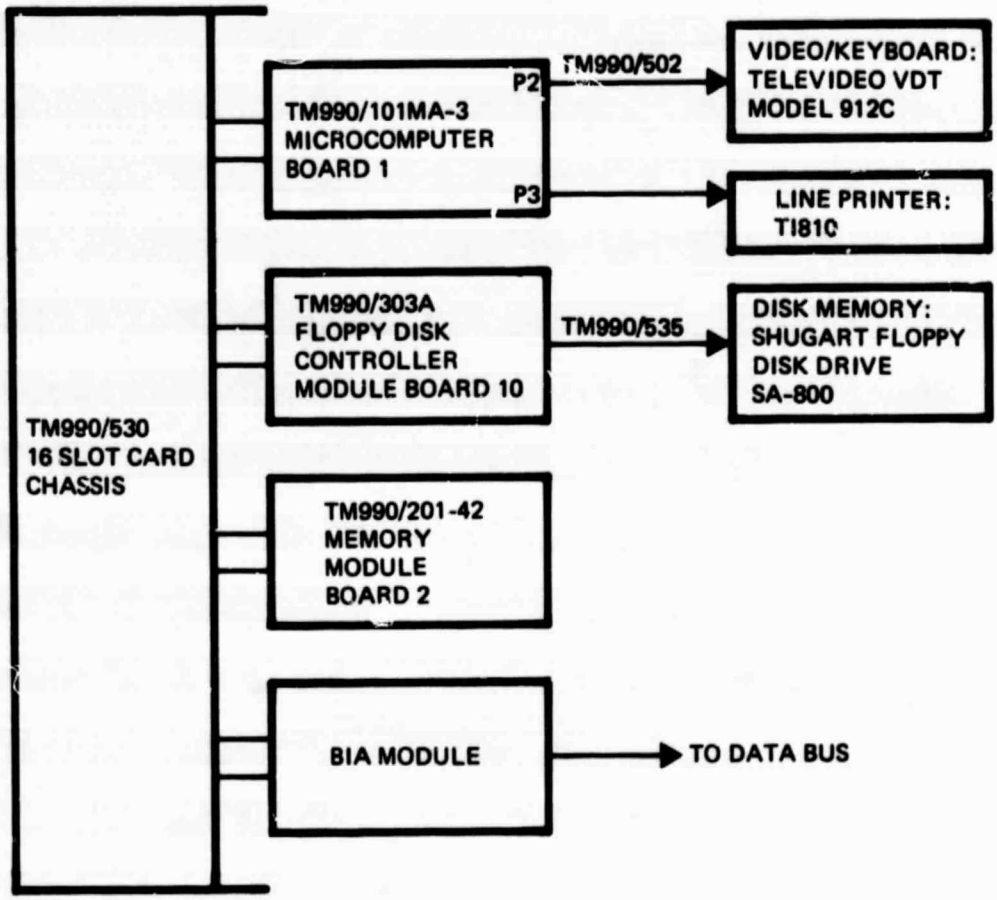


Figure 7-7. Electrical Power Subsystem Controller Inter-Rack Cables

Figure 7-8. Load Center Controller Block Diagram

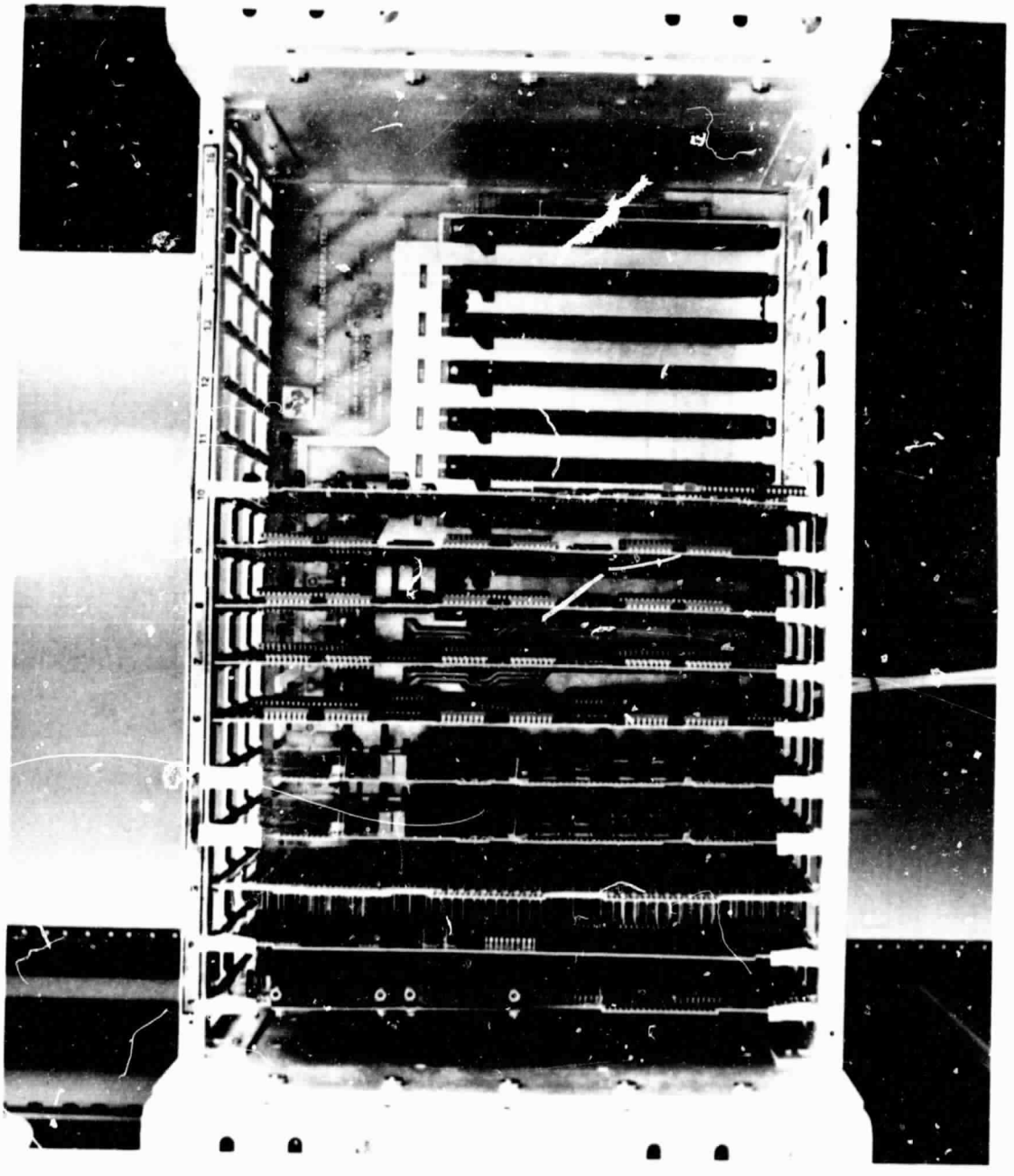


Figure 7-9. Load Center Controller (19 inch rack mounting)

The microcomputer and memory cards are the same as those in the EPS controller and are shown in Figures 7-5 and 7-6, respectively. The digital I/O, analog input, and floppy disk controller cards are shown in Figures 7-10, 7-11, and 7-12, respectively. An interconect diagram is shown in Figure 7-13. A detailed description of the printed circuit cards is contained in Appendix B.

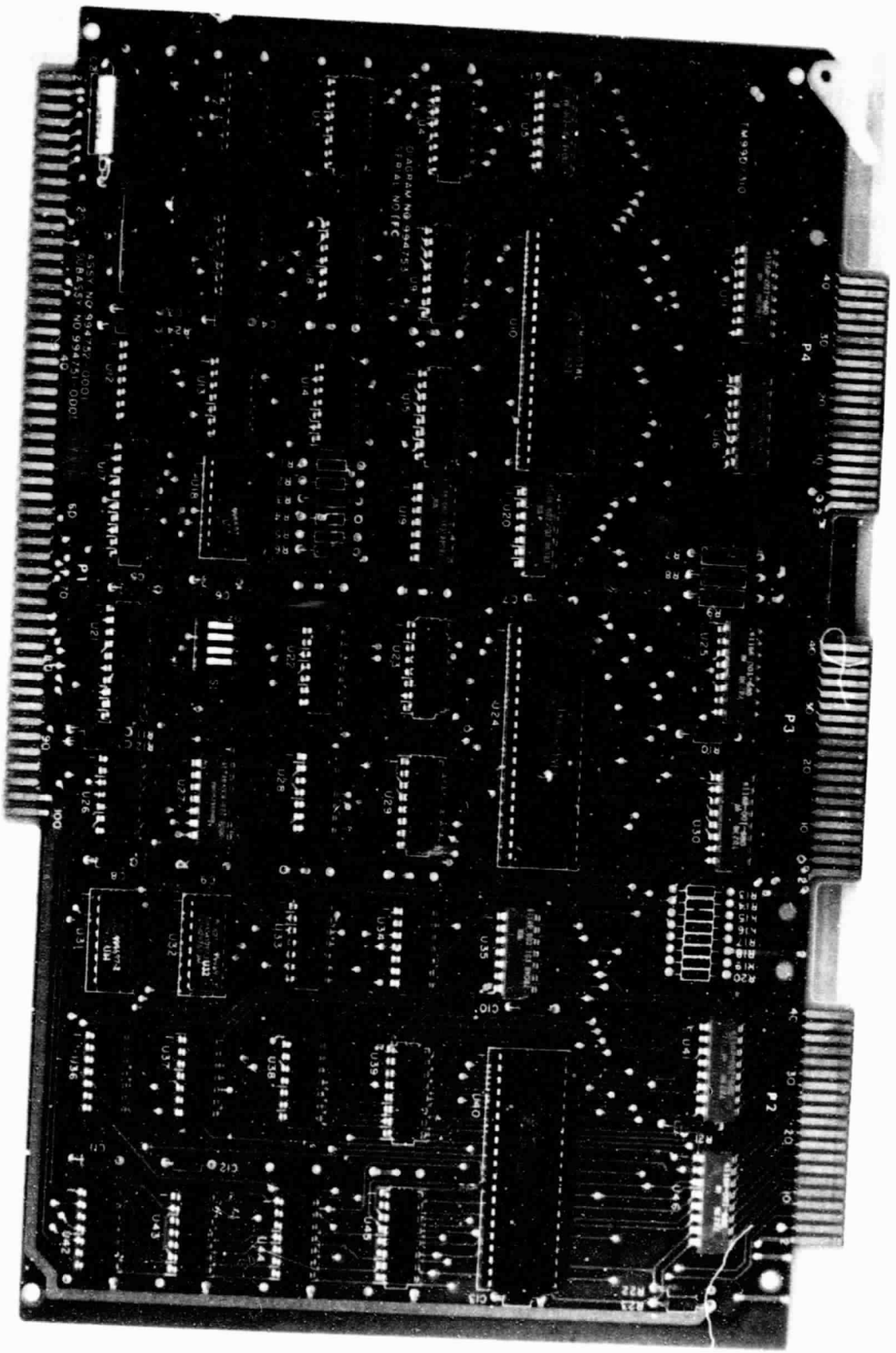


Figure 7-10. Digital Input/Output Card (7.5 x 11 inch card)

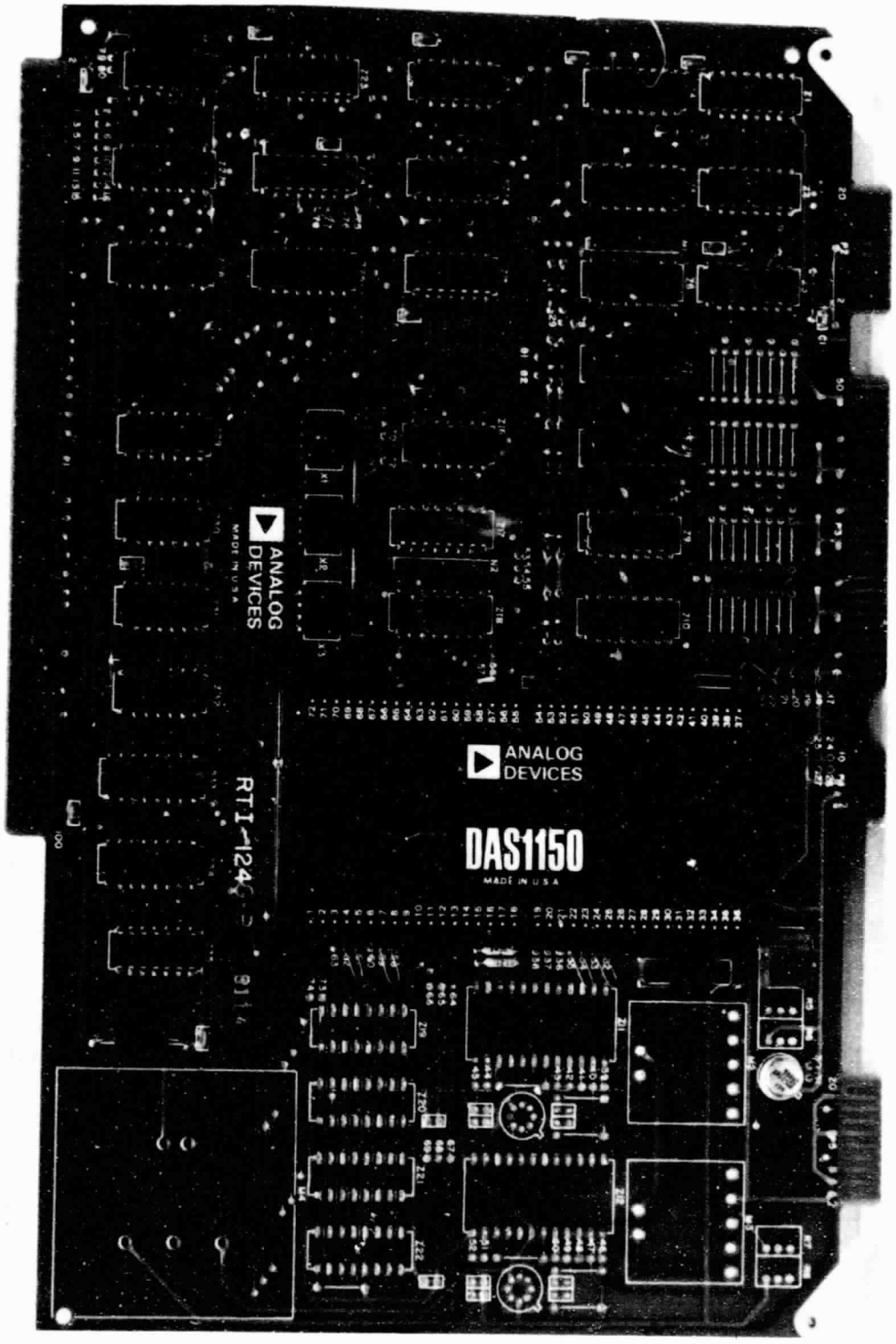


Figure 7-11. Analog Input Card (7.5 x 11 inch card)

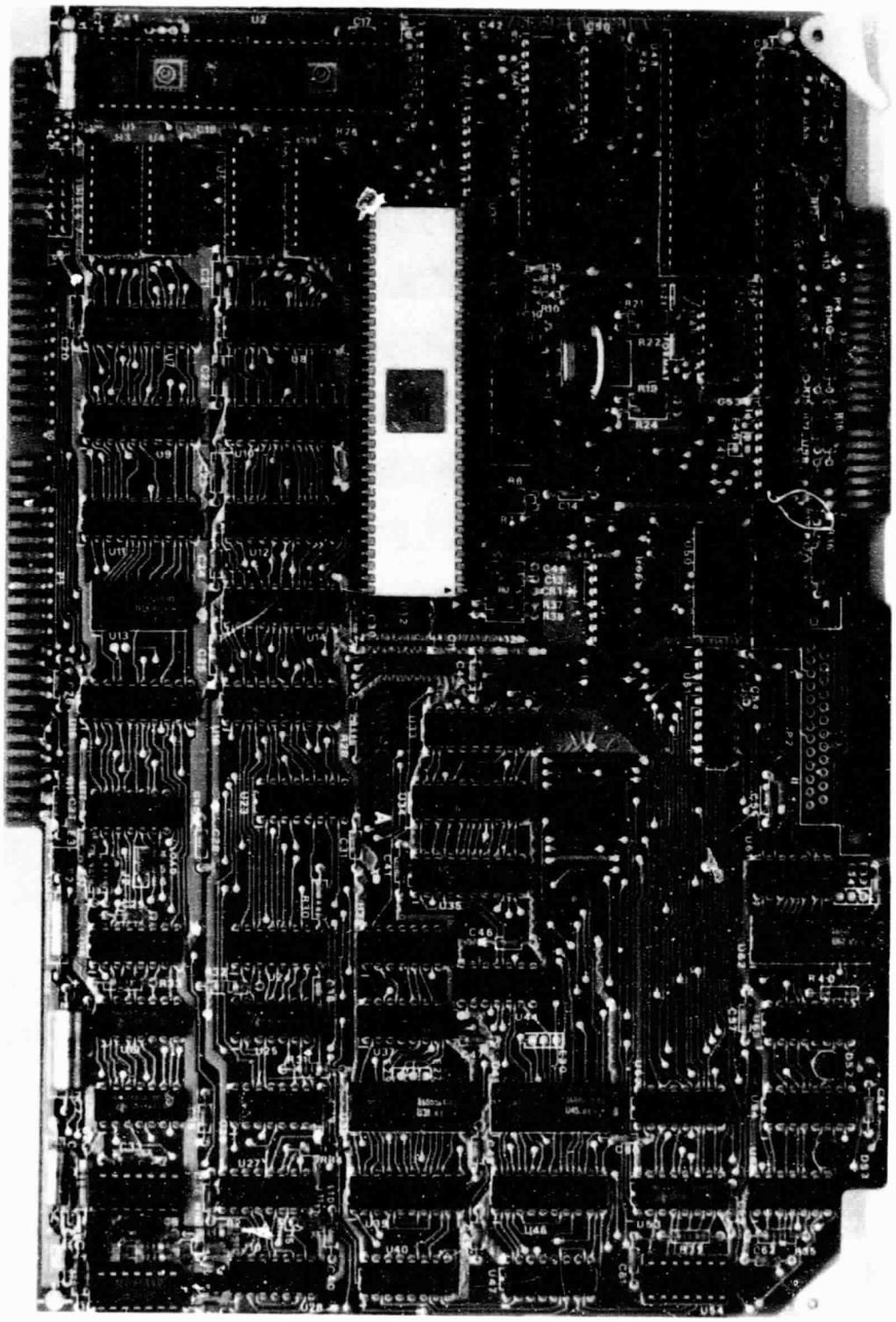


Figure 7-12. Floppy Disk Controller Card (7.5 x 11 inch card)

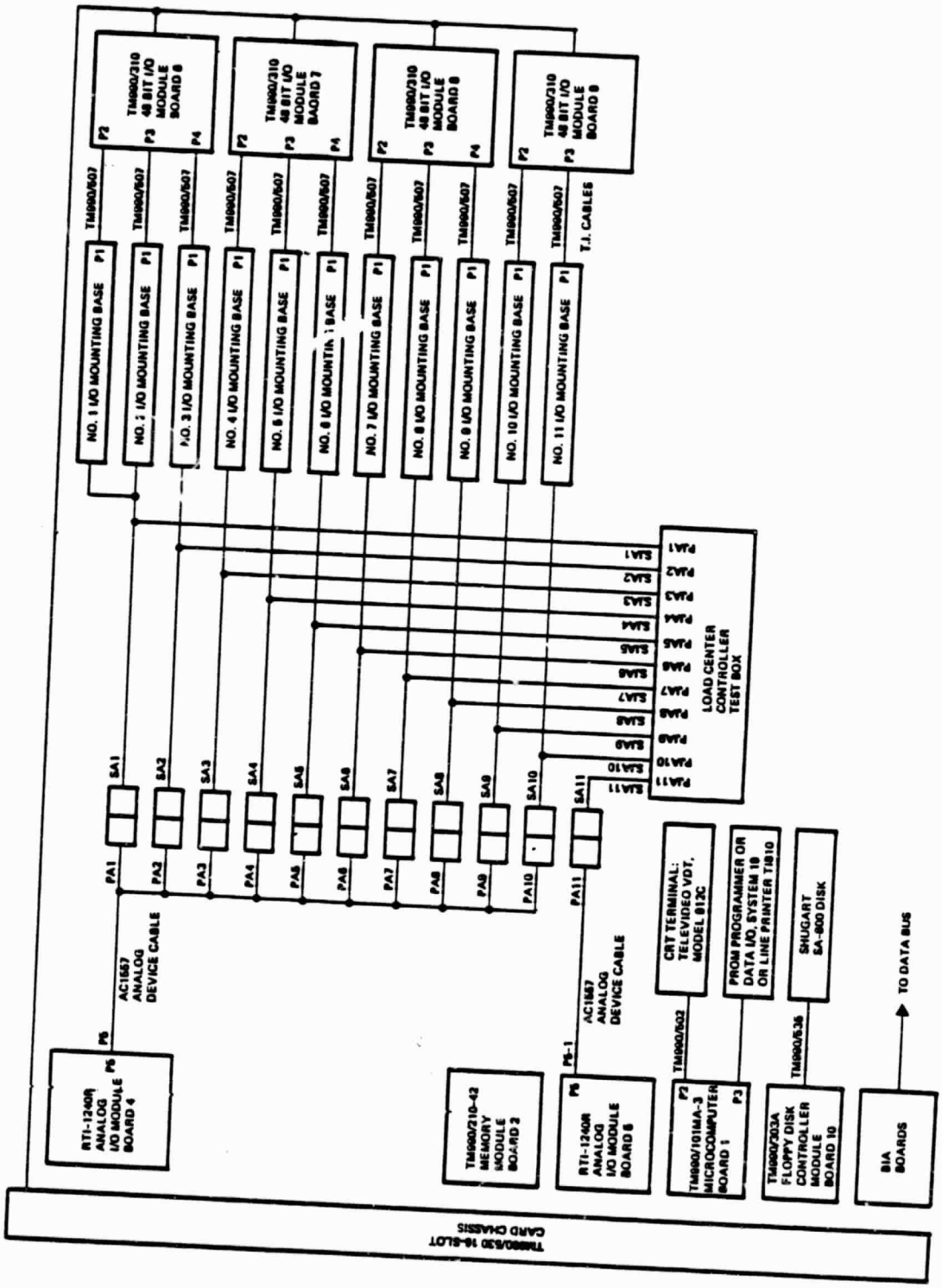


Figure 7-13. Load Center Controller Interconnect Diagram

8. TRADE STUDIES AND ANALYSES

Trade studies and analyses were performed in order to derive the appropriate power management subsystem (PMS) design. These trade studies and analyses are reported in this section.

8.1 COST SAVINGS FROM AUTONOMOUS MANAGEMENT

Analyses were performed to define the electrical power subsystem (EPS) cost savings that can be derived from autonomous management of the power subsystem. These cost analyses indicate that significant savings can result with autonomous management from reduction in ground support cost, from battery depth of discharge control (which enhances battery life), and from minimizing the operating time of the thermal coolant pumps (which lengthens pump replacement intervals) during reduced power periods.

8.1.1 Ground Station Cost Reduction

An analysis of ground station costs was performed to determine the potential for cost savings from incorporation of autonomous management of the power system. Ground station costs include the initial installation of the station, recurring maintenance and equipment refurbishment, and the support personnel to operate the station and verify satellite power system operation. A ground station similar to the Apollo tracking station at Goldstone is considered representative for the 250-kilowatt satellite. Projected cost of this station is $20M to install in 1980 (Reference 8-1). Refurbishing and equipment maintenance costs are typically 5 percent per year ($1M). The manpower to operate and maintain this ground station for the satellite power system is projected at 12 people and $910K per year, as summarized in Table 8-1.

Satellite-to-ground telemetry requirements can be reduced greatly if the power system management is autonomous (on-board) and only data for out-of-tolerance performance and equipment degradation projections are transmitted to the ground. The telemetry data is thereby reduced to that for satellite resupply and refurbishment planning functions. Thus, a separate, satellite-dedicated, telemetry-and-control station is not required. Existing communication networks, such as the Tracking and Data Relay Satellite System (TDRSS), can be utilized on a time-share basis. This mode of

Table 8-1. Ground Station Personnel Annual Cost

Function	People	Burdened Salary	Total Cost
Flight operators	2	$ 85K	$170K
Analysts	6	85K	510K
Technical Manager	1	100K	100K
Programmer	1	60K	60K
Secretary/Administrative	1	35K	35K
Clerical	1	35K	35K
Annual Personnel Cost			$910K

operation is projected as a one-man level of effort with time-share rental fees for communications. The annual cost is estimated at $100K.

The projected costs of these power system management approaches (ground station control and autonomous on-board control) are compared over a 30-year period (Figure 8-1). The aggregate cost for ground station control includes $2M for satellite telemetry transmission equipment (high data rate). The aggregate costs for autonomous management include $2M for the satellite on-board data processors. A substantial cost savings, approximately $74M over 30 years (Figure 8-1), is indicated with autonomous power management.

8.1.2 Battery Life Cycle Cost Reduction

Battery cost savings are obtained when utilizing a PMS over the life of the satellite. This analysis shows how the PMS can save on battery cost by increasing the mean life of the batteries. Potential savings are beyond $5M on the nickel-hydrogen battery system for the 250 kWe space platform with 30-year mission life. First, a formulation of present expected cycle life of a battery for various depths of discharge (DOD) is developed from the empirical data (Figure 8-2) obtained from ongoing tests. A straight

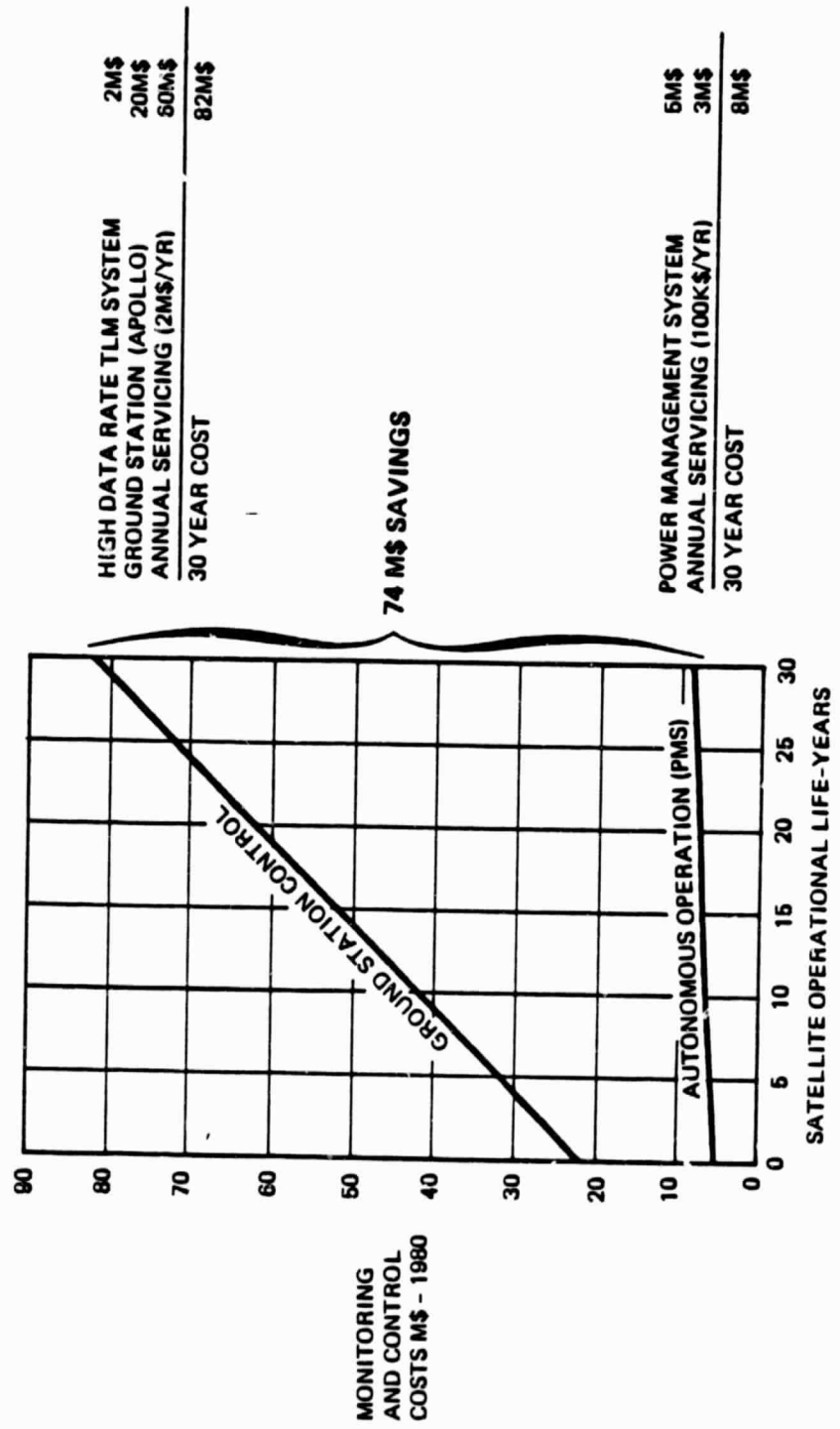


Figure 8-1. PMS Reduces Monitoring and Control Costs

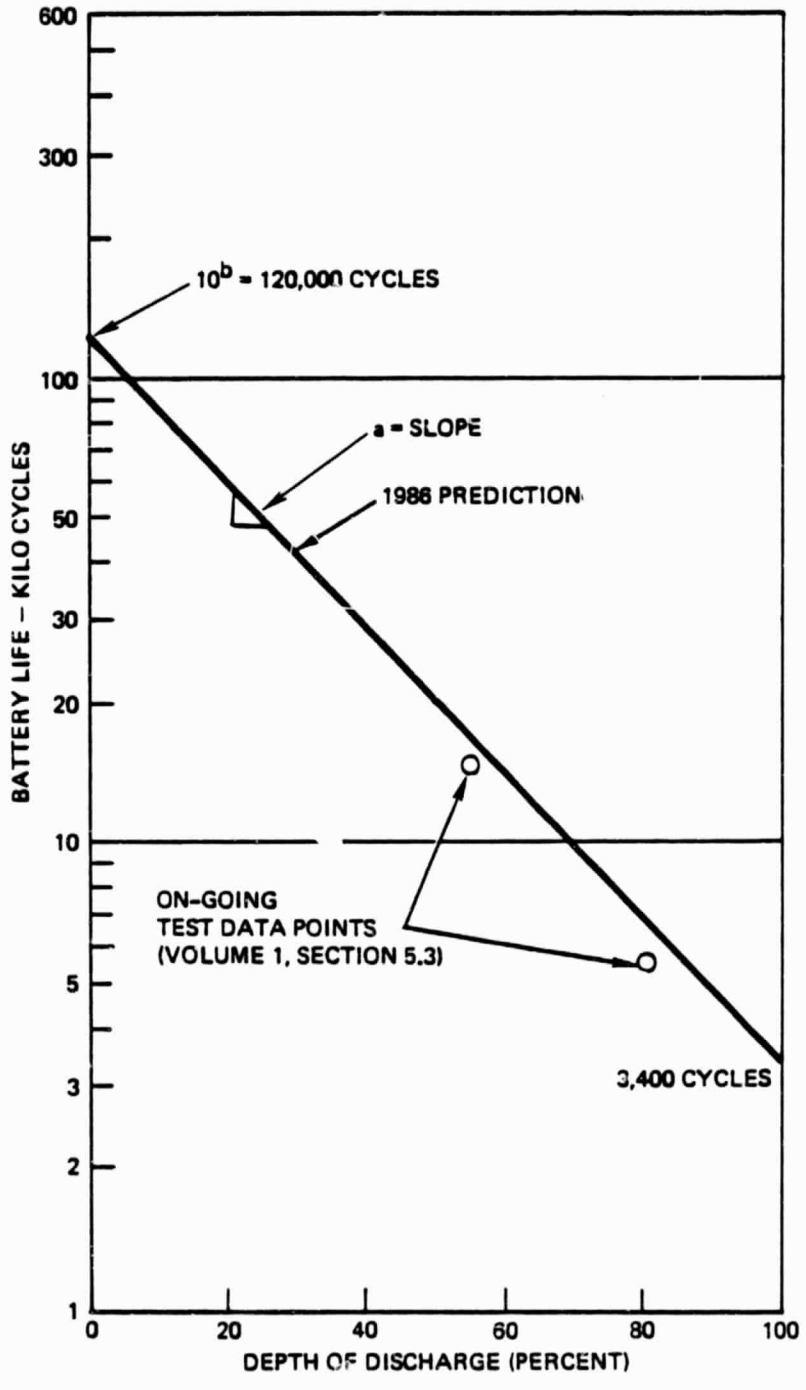


Figure 8-2. Ni-H_2 Battery Cycle Life Projection in LEO

line approximation on the semilog graph of Figure 8-2 for 1986 gives the following relationship with x = DOD.

$$\text{Cycle life} = L = 10^{b-ax}$$

From Figure 8-2:

$$b = \log 120{,}000 = 5.079$$

$$a = \log 120{,}000 - \log 3{,}400 = 1.548$$

$$L = 10^{5.079-1.548x}$$

The cost of the batteries is given by the following relationship over a 30-year life of the power system:

$$\text{Life cycle cost} = C_R \frac{30}{\text{Life}_M} + C_i$$

where

C_R = replacement cost of batteries

C_i = initial cost of batteries

Life_M = mean life of batteries

The battery life cost is minimized by maximizing the mean life. A two-battery system, shown in Figure 8-3, is used to evaluate operation of a multi-battery system with balanced or unbalanced depth of discharge to maximize the mean life of the batteries.

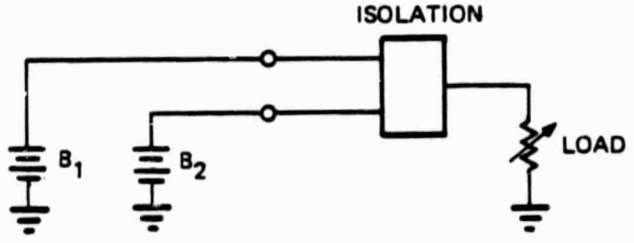


Figure 8-3. Two-Battery System

For the two-battery system shown in Figure 8-3, the following is true:

$$BC_{total} \times DOD_{design} = Load_{design}, \text{ where } BC = \text{battery capacity}$$

$$BC_1 + BC_2 = BC_{total}$$

$$DOD_1 \times BC_1 + DOD_2 \times BC_2 = \text{load (Ah)}$$

and

$$BC_1 = BC_2 \text{ by design.}$$

$$DOD_1 + DOD_2 = \text{load}/BC_1 = K \quad \text{where } K \leq 2$$

since the battery capacity is always greater than the load. This equation gives DOD_2 as a linear function of DOD_1,

$$DOD_2 = K - DOD_1$$

The total cost for the two-battery system over 30 years is given by:

$$\text{Life cost} = \frac{C_R}{2}\left(\frac{30}{L_1}\right) + \frac{C_R}{2}\left(\frac{30}{L_2}\right) + C_i$$

$$= C_R \cdot 30\left(\frac{0.5}{L_1} + \frac{0.5}{L_2}\right) + C_i$$

where L_1 and L_2 are the lives of the respective batteries.

It then follows that the mean life is determined as follows:

$$Life_M = 2\left(\frac{L_1 L_2}{L_1 + L_2}\right)$$

To minimize the life cost, the mean life must be maximized; therefore the derivative of the mean life is taken with respect to $DOD_1 = x_1$ and set equal to zero.

$$\frac{d}{dx_1} \text{Life}_M = \frac{d}{dx_1} 2\left(\frac{L_1 L_2}{L_1 + L_2}\right) = 0$$

Solving this equation, it follows that $x_1 = K/2$ gives maximum mean life. The details of this derivation are shown in Appendix E.

To illustrate the potential battery replacement cost increase for unbalanced depth of discharge operation, a graph of the increased life cost of the batteries versus the degree of unbalance assuming a 250-kilowatt load and a 175-kilowatt load is given by Figure 8-4. The depth of discharge unbalanced ratio is given by assuming that half of the batteries are operating at DOD_1, and the other half at DOD_2 where

$$\frac{DOD_1 + DOD_2}{2} = DOD_{load}$$

which is the required depth of discharge of the total battery capacity to meet the load specified. The points plotted are tabulated below:

Load (kW)	DOD_1 (%)	DOD_2 (%)	DOD_1/DOD_2	Increased Cost (M$)
250	34.0	34.0	1.00	0
250	30.2	37.8	0.80	2.8
250	24.0	44.0	0.55	9.0
250	19.4	48.6	0.40	18.5
250	14.0	54.0	0.26	31.7
175	24.0	24.0	1.00	0
175	18.0	30.0	0.60	0.7
175	14.0	34.0	0.41	2.8
175	8.0	40.0	0.20	11.7

A straight line approximation on a semilog plot for the cycle life of a battery is thought of as a conservative estimate by battery experts for low depth of discharge. It is felt that with a cycle life for a very low depth of discharge, a much greater battery life will be realized than that projected herein. To illustrate how this increased life expectancy might

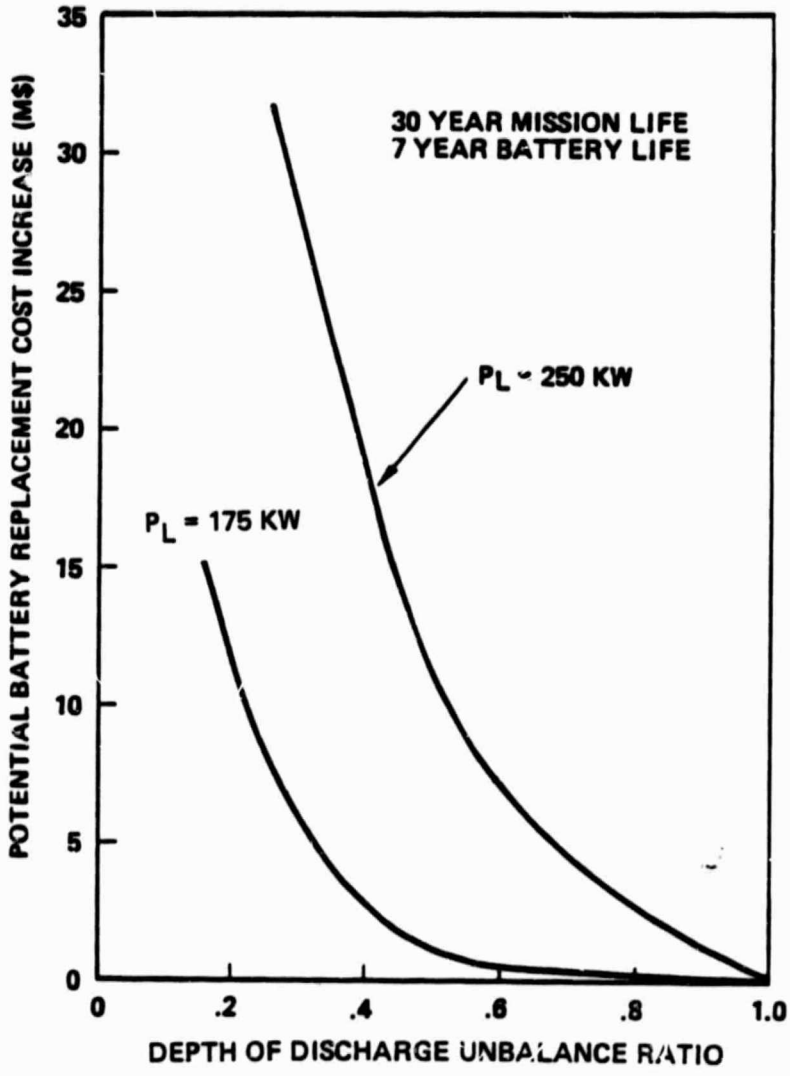


Figure 3-4. Battery Load Balancing Reduces Cost

affect the previous derivation, another approximation of the cycle life versus depth of discharge relationship is used which gives infinite life at zero percent depth of discharge. That is,

$$L = \frac{10^{b-ax}}{x}, \quad 0 < X \leq 1$$

Using this relationship, it can be shown that the maximum of the battery mean life for a two-battery system is still obtained by having a balanced depth of discharge, that is $x_1 = x_2 = K/2$.

If the cycle life has an extreme nonlinearity near a depth of discharge required to meet that load demand for balanced operation, it is possible that a more optimal distribution of battery usage may be found. This might be the case for a lightly loaded system when the cycle life versus depth of discharge plot appears as shown in Figure 8-5.

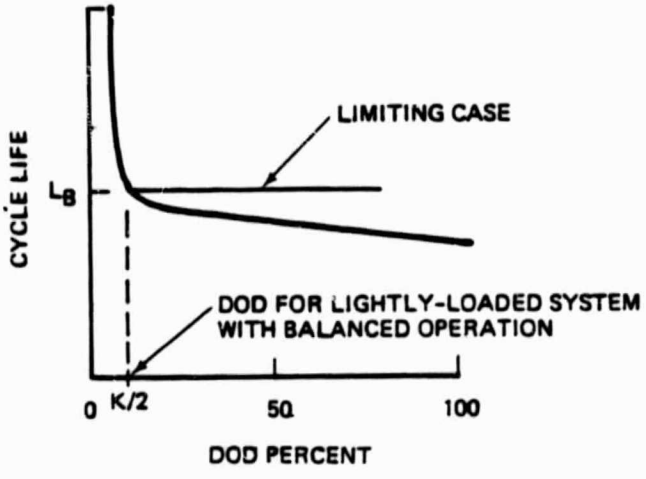


Figure 8-5. Life/DOD Relationship Cycle of a Battery with Extreme Nonlinearity Near Balanced DOD

To illustrate this, the two-battery system will again be used. The mean life for balanced operation is simply the cycle life of a single battery since $L_1 = L_2 = Life_M$:

$$Life_M = 2\left(\frac{L_1 L_2}{L_1 + L_2}\right)\Bigg|_{L_1 = L_2} = L_1 = L_2$$

Then letting L_B = mean cycle life of the batteries for the depth of dis

charge which gives balanced operation, the following relationships are used to represent the nonlinearity in the cycle life curve:

$$L_1 = \frac{L_B}{K_1} ; \quad L_2 = K_2 L_B ; \quad 1 \leq K_1, K_2$$

x_1 and x_2 give L_1 and L_2 respectively with $x_1 + x_2 = K$ and K_1 and K_2 are undefined functions of x_1. Then, the ratio of the mean life for unbalanced (L_{uB}) operation to balanced operation (L_B) gives the life ratio:

$$\frac{L_{ub}}{L_B} = \frac{2\left(\frac{L_1 L_2}{L_1 + L_2}\right)}{L_B} = \frac{2 K_2}{1 + K_1 K_2}$$

For the limiting case where $K_1 = 1$ and K_2 , the life ratio approaches 2. For this case where $L_1 = L_B$ for increasing DOD, as is illustrated in Figure 8-5, DOD_2 is reduced to a point where L_2 is infinite. (Note: Battery 1 and Battery 2 have identical cycle life/DOD relationships.) As a result twice the mean life of balanced operation is obtained for this unbalanced case. This is equivalent to having B_1 solely supply the load and letting B_2 rest where B_1 has the same cycle life expectation whether solely supplying the load or equally sharing it with B_2. It can be further shown that for an N battery system the limit ratio is N. These limiting cases are nonrealizable since $K_1 \neq 1$ for increasing DOD_1, but potential savings exist if $K_1 < 2$ and $K_2 > 1/2\ (1 + K_1 K_2)$ for the two-battery system. A similar relationship can be derived for an N battery system.

In conclusion, we see that the battery system should be operated with balanced DODs to substantially reduce cost. However, potential savings may also be obtained in battery costs if some distinct nonlinearities exist in the cycle life/DOD relationship by operating the battery system with a more optimal distribution of depth of discharge.

8.1.3 Thermal Coolant Pump Life Cycle Cost Reduction

The power management subsystem can also reduce life cycle costs by extending the interval between replacement of thermal coolant pumps. This

is accomplished by shutdowns of some pumps during periods of reduced electrical loads. For example, consider the payload power utilization of Figure 8-6. During periods of reduced electrical load, the thermal cooling requirements are directly reduced, coolant flow can be managed accordingly (rather than relying wholly on bypass valves), and some pumps can be shut down:

20 pumps are used for 250 kilowatts, 25 percent of the time;
16 pumps are used for 200 kilowatts, 50 percent of the time;
12 pumps are used for 150 kilowatts, 25 percent of the time.

The average is only 16 pumps in operation and reduces the pump replacement costs by 5.3M$:

$$20 \text{ pumps} \times 100\text{K\$/pump} \times \frac{30 \text{ yr} \times 8766 \text{ hr/yr}}{20{,}000 \text{ hr}} = 26.3\text{M\$}$$

$$16 \text{ pumps} \times 100\text{K\$/pump} \times \frac{30 \text{ yr} \times 8766 \text{ hr/yr}}{20{,}000 \text{ hr}} = 21.0\text{M\$}$$

$$\text{Potential cost savings} \qquad 5.3\text{M\$}$$

The degree to which this potential cost savings can be realized is dependent upon the judicious layout and organization of the coolant plumbing consistent with the payload operating scenarios. The resultant complexity of the coolant circuits must be consistent with the possible cost reduction in coolant pump life cycle costs. For example, several small pumps per coolant loop may be more cost effective than one large pump.

8.2 POWER MANAGEMENT SYSTEM CONTROL HIERARCHY

Tradeoff studies were performed to determine the control hierarchy for the power management subsystem (PMS) within the electrical power system. Centralized, distributed, and several hybrid concepts were considered (Table 8-2). The three-tier hybrid concept (Figure 8-7) was selected as the recommended hierarchy. This selection requires modest computational power in each controller, provides a well defined interface with the spacecraft computer and other subsystems, provides subsystem autonomy for assembly, testing, and algorithm development, minimizes the quantity and distance of sensor data transfer, incorporates subsystem control to simplify controller interactions and communications, and accommodates growth in both

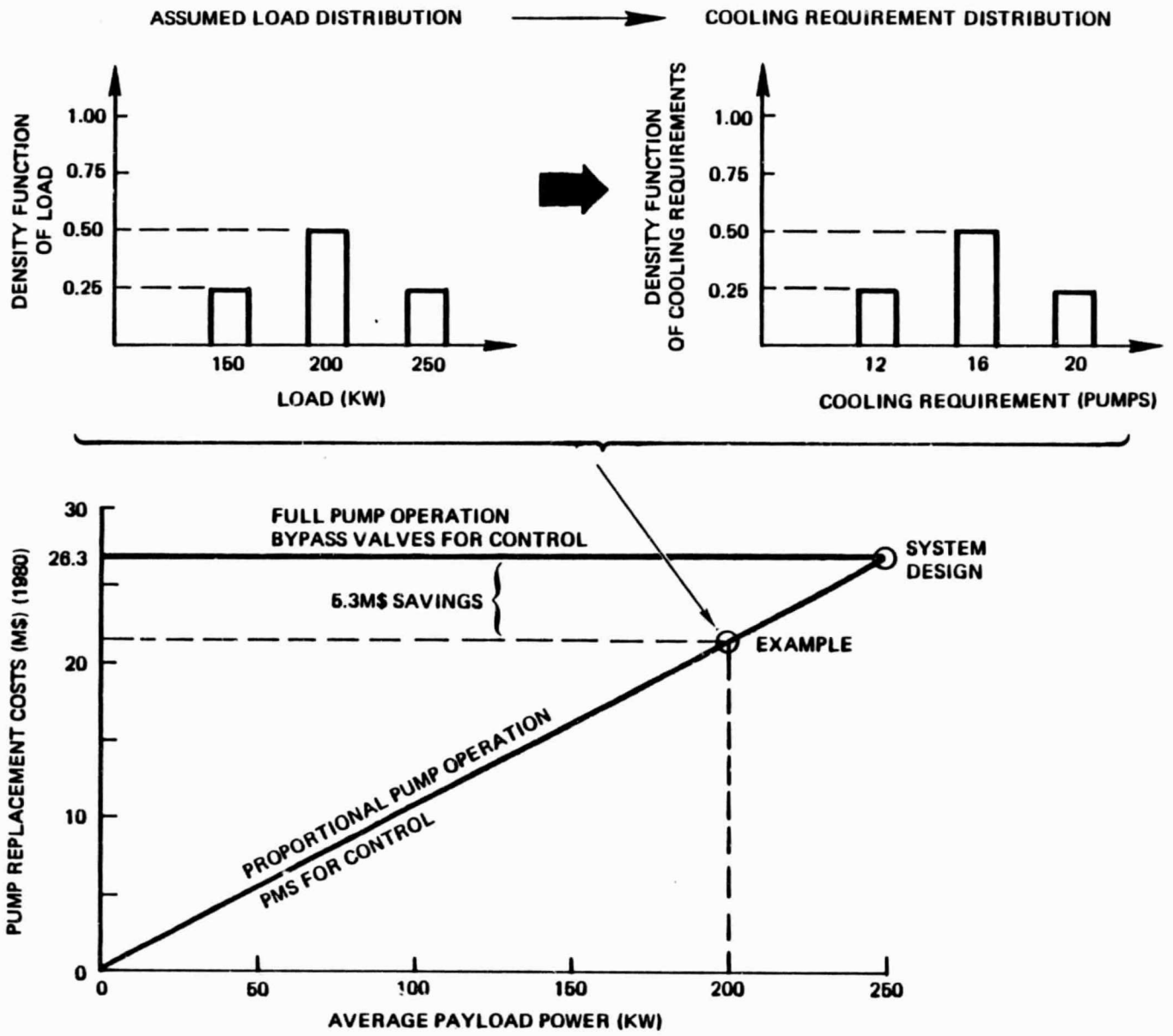
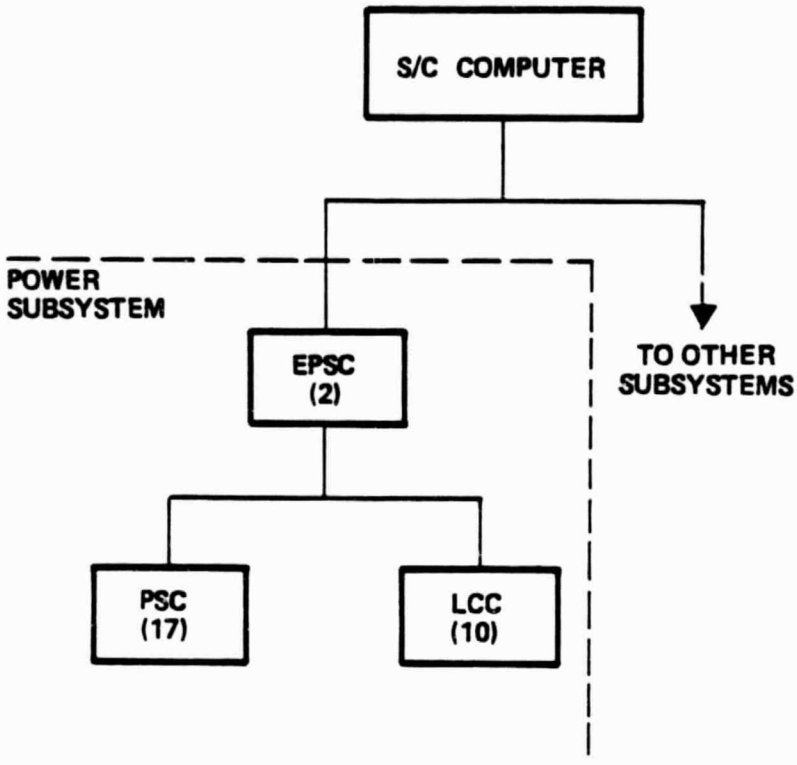


Figure 8-6. Autonomous Power Management Can Reduce Coolant Pump Costs

Table 8-2. Control Hierarchy Selection

Description	Tier Levels	Figure	Microprocessor/Computer Locations			Comments	Selection
			Module	Subsystem	Spacecraft		
Distributed module controllers	1	8-10	✓			Very complex software	
Spacecraft and module controllers	2	8-9	✓			More costly development	Alternate 2
Subsystem and module controller	2	8-8	✓	✓		Complex software	Alternate 1
Spacecraft, subsystem and module controllers	3	8-7	✓	✓	✓	Well defined; early demo	Recommended
Distributed subsystem controllers	1	8-13		✓		Large EPS computer; complex software	
Spacecraft and subsystem controller	2	8-12		✓	✓	Large subsystem computer	
Centralized spacecraft controller	1	8-11			✓	Very large computer; costly software	



EPSC = ELECTRICAL POWER SUBSYSTEM CONTROLLER
PSC = POWER SOURCE CONTROLLER
LCC = LOAD CENTER CONTROLLER

Figure 8-7. Recommended PMS Architecture (Spacecraft, Subsystem, and Module Controllers)

energy storage (number of channels) and load center quantity. In addition, the algorithms, control routines, and controller hardware developed for this control hierarchy are applicable to the alternate approaches of Table 8-2.

The recommended approach (Figure 8-7) employs three controller types: power source controller (PSC), load center controller (LCC), and electrical power subsystem controller (EPSC). This division of the control hierarchy is based upon the inherent separation of performance functions and their related spacecraft geographic locations. Sensor data transfer beyond the geographically separated local controllers is essentially eliminated. Communication between controllers occurs with derived data rather than sensor data. The controller-to-controller communication (data bus) traffic is thereby minimized and simplified. Further this natural division of control functions enables easy growth of the PMS with electrical power level growth by the addition of a power source controller with each additional battery/channel and by the addition of a load center controller with each additional load center.

Alternate 1 configuration (Figure 8-8) omits the spacecraft computer. The spacecraft computer functions are distributed as appropriate into the controllers of the various spacecraft subsystems. This approach has essentially all of the performance attributes of the recommended option (Figure 8-7). However, a significant degree of complexity is added to the subsystem controller interface software in order to communicate among subsystem controllers over the spacecraft data bus without a centralized controller.

Alternate 2 configuration (Figure 8-9) retains the spacecraft computer, but eliminates the subsystem level controller. The functions of the subsystem level controller are transferred to the spacecraft computer. This approach retains most of the attributes of the recommended option (Figure 8-7) except for independent subsystem testing and subsystem control algorithm implementation. With subsystem control functions performed by the spacecraft computer, a more complex human interface is required to develop, code, and implement these subsystem control algorithms. This will increase software costs. Also, the spacecraft computer (or equivalent subsystem support equipment - an additional cost) is required for subsystem level integration testing. In addition, the data bus interface for the

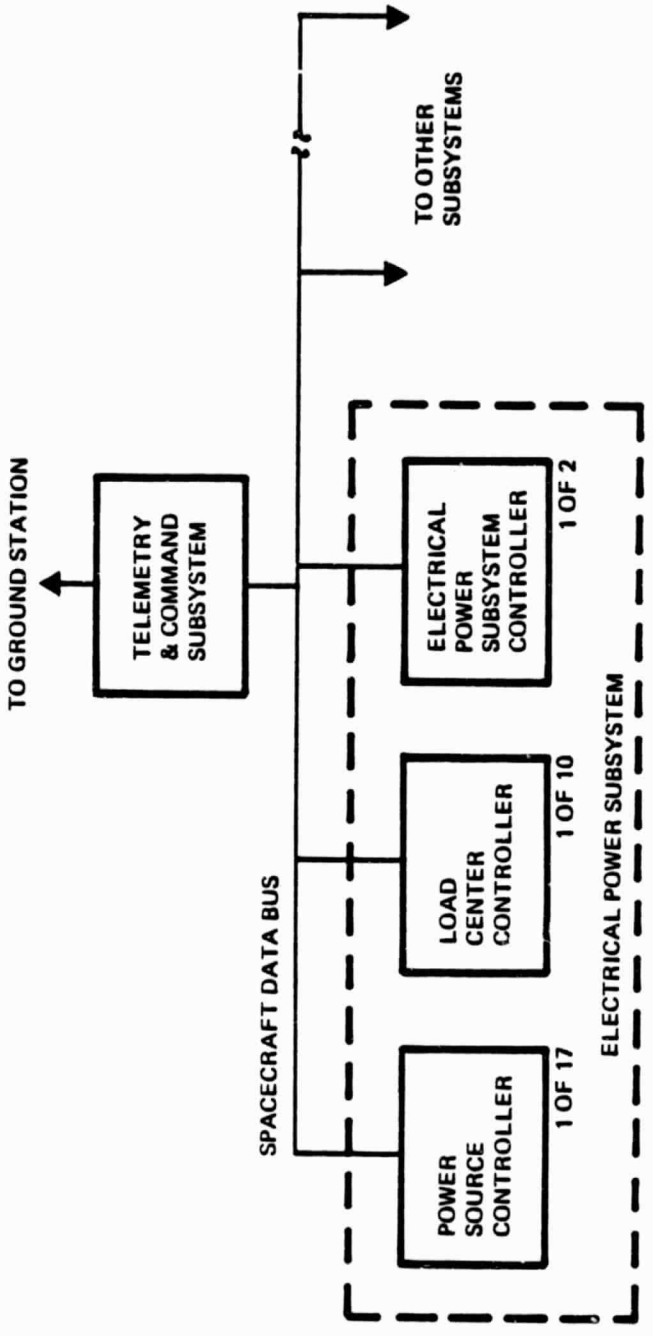
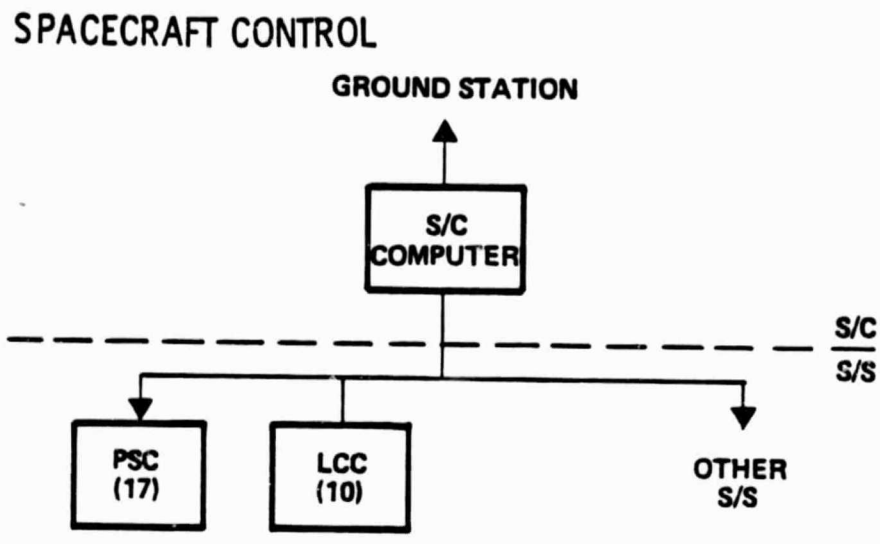


Figure 8-8. Alternate 1 PMS Architecture (Subsystem and Module Controllers)



ADVANTAGES

- EASIER TO IMPLEMENT REDUNDANCY

DISADVANTAGES

- POWER SYSTEM IS NOT SELF CONTAINED
- HIGH SPEED BUS INTERFACE
- EXPANSION LIMITED BY S/C COMPUTER CAPABILITY

Figure 8-9. Alternate 2 PMS Architecture (Subsystem and Module Controllers)

module level controllers will be more complex due to the higher data rates required at the spacecraft level compared to the data rate at the electrical power subsystem level of the recommended approach.

The subsystem-distributed architecture (Figure 8-10) employs only module level controllers. Consequently executive control functions must be distributed into these module level controllers. The resulting software to control communications on the data bus is very complex because executive control passes from controller to controller at this module level. This software complexity is expected to increase software costs in excess of any hardware savings in avoiding the subsystem level controller or spacecraft computer.

Three approaches avoid module level controllers: centralized spacecraft control (Figure 8-11), spacecraft and subsystem controller architecture (Figure 8-12), and distributed subsystem controller architecture (Figure 8-13). Two of these require a large electrical subsystem computer capability (Table 8-3), and the third requires a very large computer for spacecraft centralized control (Figure 8-11). However, each sensor geographic area is serviced by "dumb" data bus interface adapters. Hence, little cost savings are realized in hardware with these approaches. In addition, a high data rate communication bus is required to transmit this sensor data to the centralized processing locations. Also, growth is limited by the initially installed computational and memory capacities. Therefore, these approaches are rejected in favor of local (module level) processing options.

8.3 DATA NETWORK ARCHITECTURE

A global bus architecture was selected as the data bus concept for the power management system because of its flexibility, reliability, failure tolerance, data rate accommodation, and equipment economy. This architecture uses a single twisted pair electrical path (or single optical path) as shown in Figure 8-14 to minimize wiring between controllers. In this scheme, any single controller can communicate directly with any other controller on the data bus. Broadcast messages to all controllers can also be transmitted. The architecture works in conjunction with a distributed-controlled, time sequential data bus contention resolution scheme (described in Section 8.5) and the International Standards Organization

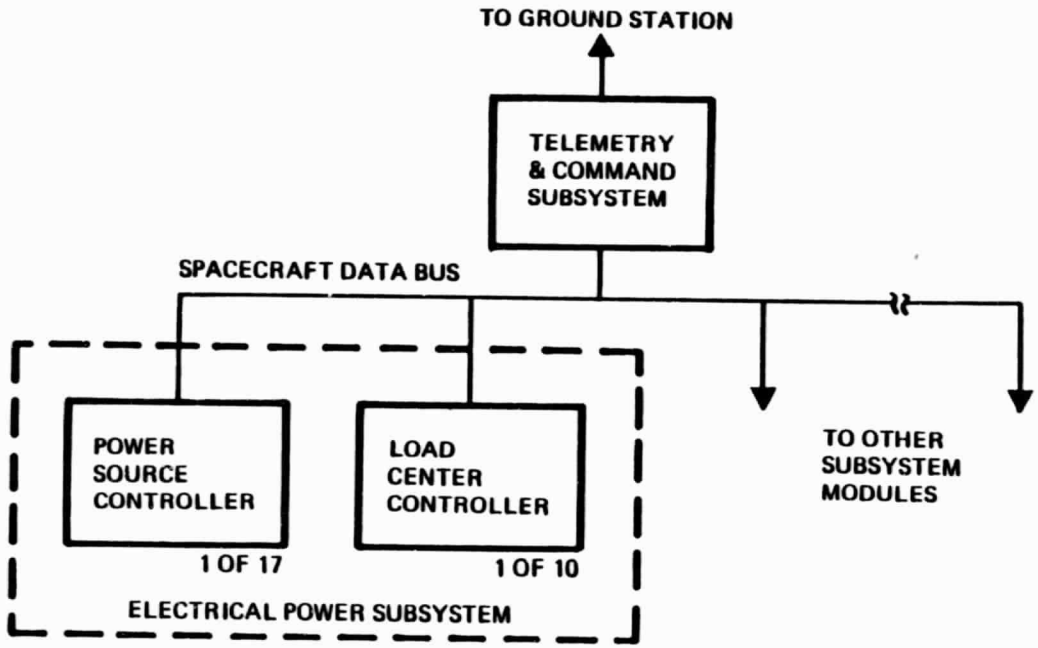


Figure 8-10. Subsystem Distributed Architecture (Distributed Module Controllers)

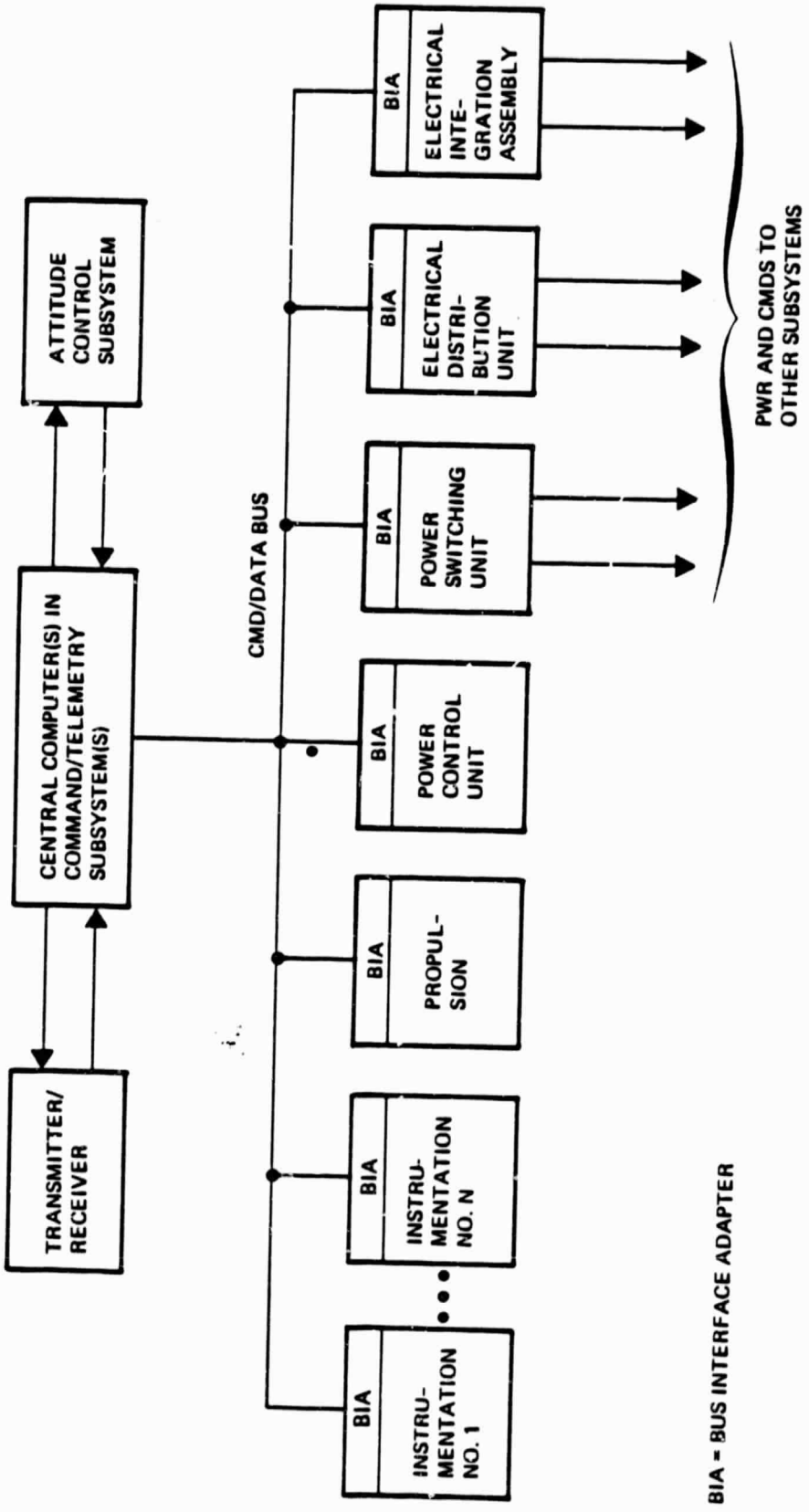


Figure 8-11. Centralized Spacecraft Control Architecture

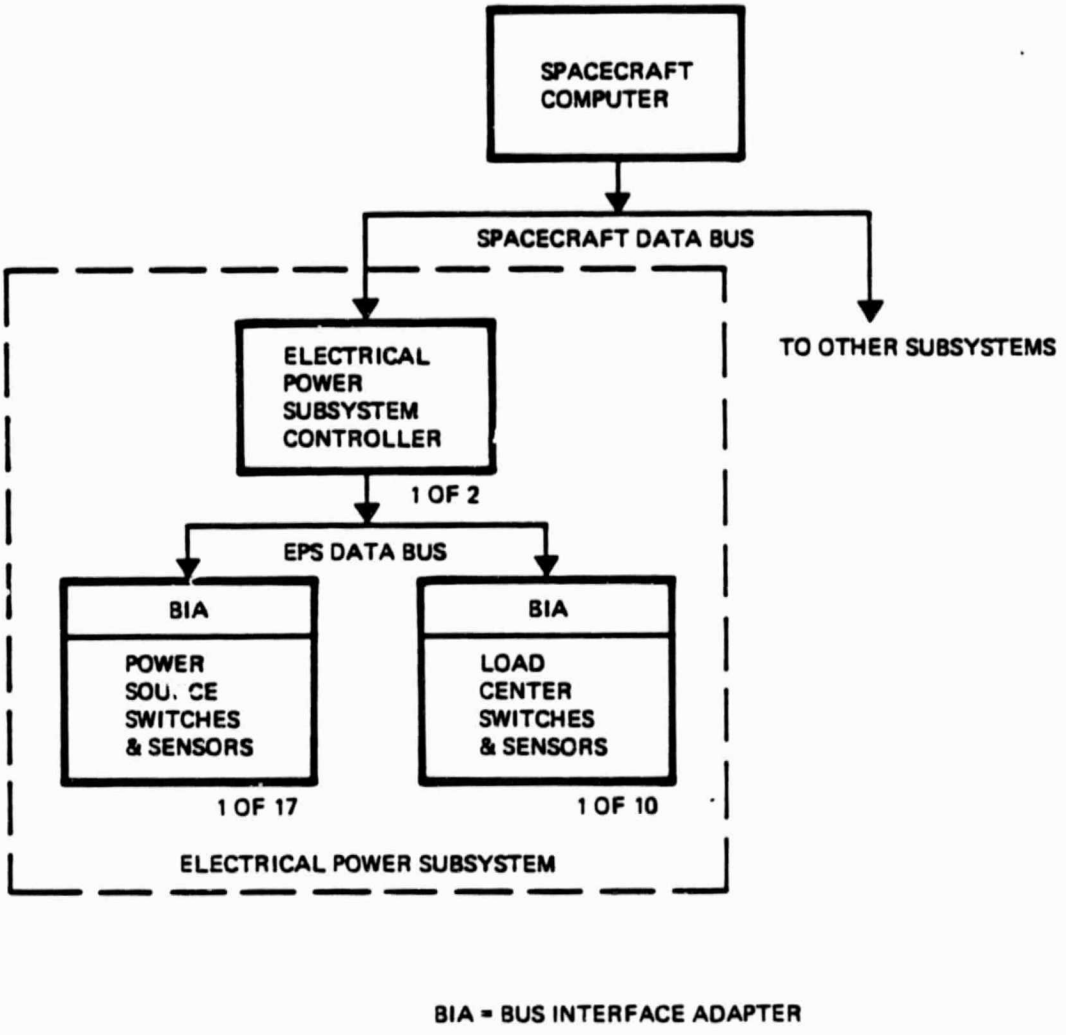


Figure 8-12. Spacecraft and Subsystem Controller Architecture

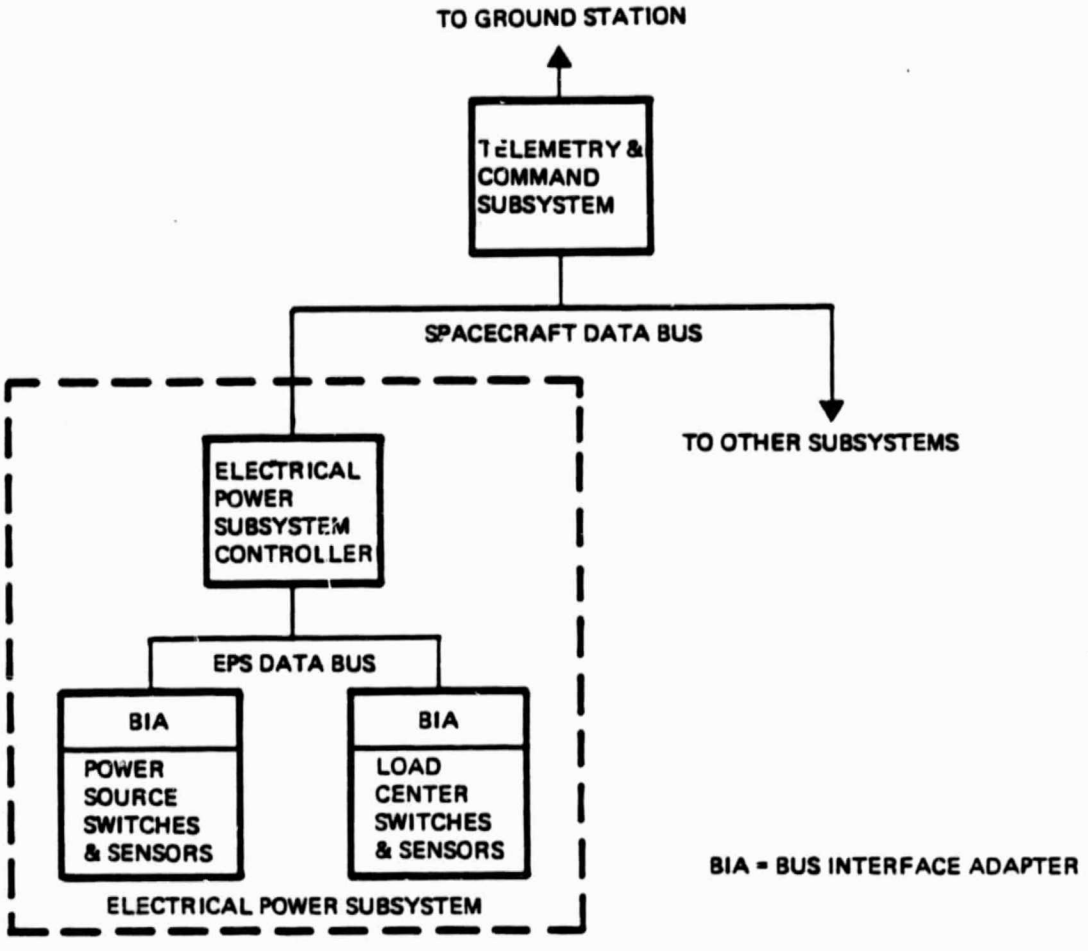


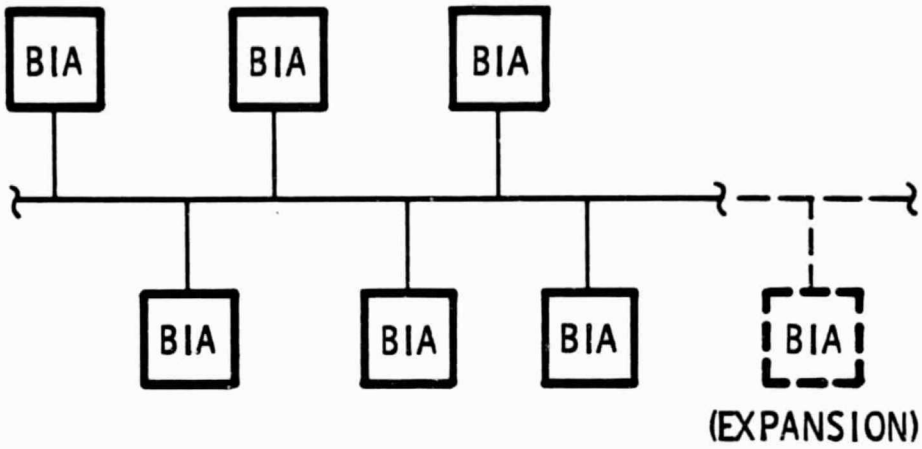
Figure 8-13. Distributed Subsystem Controller Architecture

Table 8-3. High Speed Computer Required for Centralized Approach

Functions	Repeat Time (sec)	Number of Instructions	Rate: Instructions per Second
Power Source (17)			
• Battery State-of-Health	20	83,100	71,000
• Battery Charge Control	2	7,100	60,000
• Solar Array Status	20	3,500	3,000
Load Center (10)			
• Switch and Load Monitoring and Fault Interrogation	2	17,000	85,000
• Load On/Off Command Processing	2	2,600	13,000
Power Subsystem			
• Spacecraft and Executive	1	8,700	8,700
• Power Subsystem State of Health	60	103,000	1,700
• Energy Planning and Allocation	300	8,000	50
• Load Assignments	300	5,400,000	18,000
Total Instruction Rate			260,000

Indicated Computer Speed Required for Centralized Computation:

Instructions per Second	x	30% Overhead	x	100% Margin	x	Clock Cycles per Instruction	=	Clock Cycles per Second
260,000	x	1.3	x	2	x	10	=	7 MHz



ADVANTAGES

- MINIMIZES SINGLE POINT FAILURES
- EASILY EXPANDED
- COMMON INTERFACE UNIT
- FAILURE TOLERANCE
- DATA RATE RANGE
- ELECTRICAL OR OPTICAL BUS
- FLEXIBLE, RELIABLE, ECONOMICAL

Figure 8-14. Global Data Bus Architecture

(ISO) high-level data link control (HDLC) data bus protocol (described in Paragraph 8.4) to provide the overall data communications design.

The global architecture is inherently flexible. Growth of the data bus system and/or modification of the system topology is easily accomplished by simple addition (or deletion) of the related BIAs and the associated data bus length without disturbing existing (or other) BIAs and data bus routing.

The global architecture approach, incorporating the contention resolution scheme and the frame checking sequence of the ISO HDLC protocol, is tolerant to many equipment (BIA) failures. Potential shorting of the data bus is minimized by transformer coupling between the BIA electronic components and the data bus lines. (This transformer is a critical element, though passive, and must be designed, manufactured, and tested to preclude failure). Failure of a BIA to transmit is tolerated by accommodation in the contention resolution scheme. Transmission of a garbled or unintelligible message may occur, but the message is negated at the receivers by the frame checking sequence validation routine of the ISO HDLC protocol. The following potential failure modes in a BIA affect the data bus operation:

a) Failure to terminate transmission

b) Transmission at the wrong time

c) Transmission with a wrong but validated (by the frame check sequence, FCS) transmitter address.

Assuring termination of transmission requires hardware redundancy in the BIA to preclude a single point failure. Transmission with a wrong but validated (by FCS) transmitter address causes some BIAs to be skipped in the contention resolution. Transmission by a BIA out of turn (at the wrong time) will garble the message from another BIA. Software and/or hardware protection for both these failures may be provided in the BIA design to avoid single point failures leading to these results. Alternatively, and/or in addition, the EPS controller (or other authority) may monitor the data bus for these failure signatures (skipped BIAs in the contention sequence, and/or repetitively garbled messages), and the offending BIA placed in the IDLE or RECEIVE modes until repaired or replaced.

Typical data bus bit rates are 250 kilohertz, 1 megahertz, 10 megahertz, and 40 megahertz. The lower rate (250 kilohertz) is suitable for a small power management subsystem comprising a few controllers. A 1-megahertz data rate is appropriate for the 250-kilowatt electrical power system comprising 17 power source controllers, 10 load center controllers, and redundant (two) electrical power system controllers. Larger power management subsystems or data bus commonality with other subsystems would require the higher data rates (10 to 40 megahertz). The global data bus architecture accommodates this great range of data rates. Typically, the 250 kHz to 10 MHz data rates are suitable with either an electrical wire (twisted pair) or optical data bus implementation. Higher rates, typically 40 MHz, are provided by an optical data bus path. The basic BIA design for the global data bus is compatible with each of these data rates or data bus types. The BIA bus transceiver element is modified for electrical or optical bus transmission, and the clock rate is adjusted accordingly. The data buffer in the BIA between the bus protocol control logic and the processor input/ output logic reconciles the differing data processing and transfer rates of the controller processor and the data bus.

Other data transfer architectures considered were:

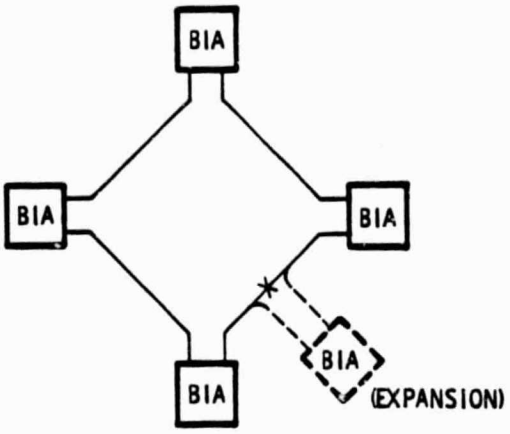
a) Ring or loop bus, Figure 8-15

b) Central switch control bus, Figure 8-16

c) Central memory bus, Figure 8-17

d) Complete interconnection, Figure 8-18.

Each of these concepts has a major deficiency compared to the selected global data bus architecture. The central switch (Figure 8-16) and central memory (Figure 8-17) architectures each have major single point failure modes in the centralized switching and memory elements. The complete interconnection architecture approach (Figure 8-18) is inflexible for application to a utility power system expecting significant growth and reconfiguration. Redesign of all the BIAs is required at each expansion, or major initial overdesign of each BIA is required. The ring or loop bus architecture (Figure 8-15) places each BIA in series with the transmitted data. This results in a simple contention resolution scheme. However, each BIA must receive and retransmit the message even if not addressed to



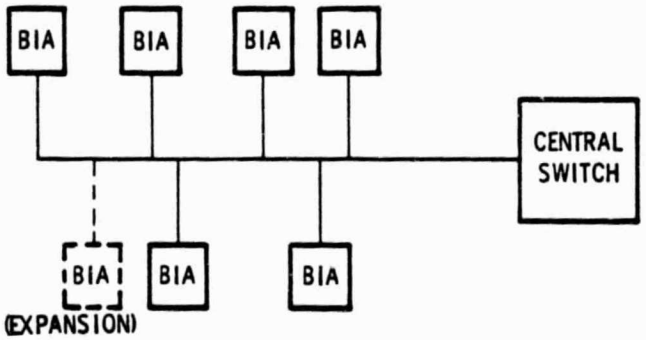
ADVANTAGES

- BUS INTERFACE UNIT (BIA) MODULARITY
- SYNCHRONIZATION SIMPLICITY
- CONTENTION ARBITRATION SIMPLE
- EXPANDABLE SYSTEM

DISADVANTAGES

- MANY FAILURE POINTS (EACH BIA)
- PROPAGATION DELAY IN EACH BIA

Figure 8-15. Ring or Loop Bus Architecture



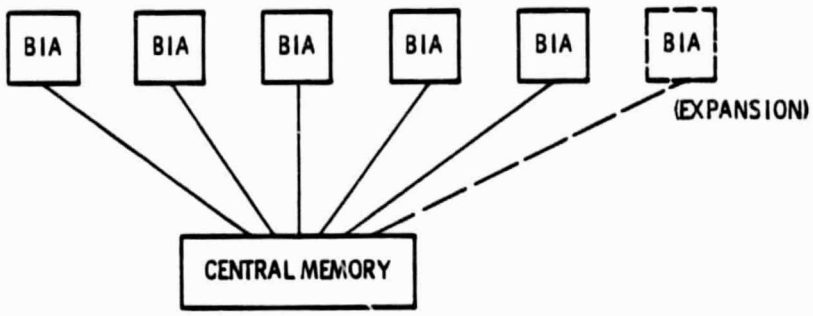
ADVANTAGES

- COMMON INTERFACE UNIT
- EASY EXPANSION
- SIMULTANEOUS MULTIPLE MESSAGES

DISADVANTAGES

- SINGLE POINT FAILURE (CENTRAL SWITCH)

Figure 8-16. Central Switch Control Bus Architecture



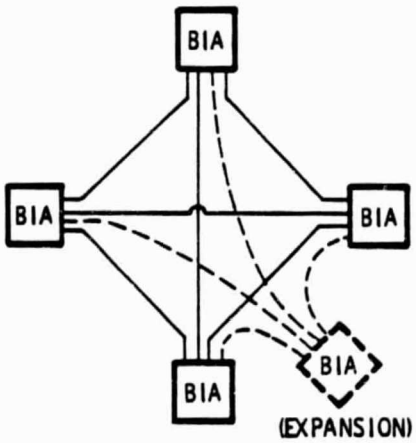
ADVANTAGES

COMMON INTERFACE UNIT
EASY BIA ADDITION

DISADVANTAGES

SINGLE POINT FAILURE MODE (CENTRAL MEMORY)
MEMORY CAPACITY LIMITS EXPANSION
HIGH MEMORY PROCESSING RATE REQUIRED

Figure 8-17. Central Memory Bus Architecture



ADVANTAGES

- AVOIDS SINGLE POINT SYSTEM FAILURES
- RECONFIGURATION CAPABILTIY
- SIMULTANEOUS MULTIPLE MESSAGES

DISADVANTAGES

- EXPANSION/GROWTH AFFECTS EXISTING UNITS
- EXTENSIVE WIRING FOR LARGE QUANTITY OF UNITS

Figure 8-18. Complete Interconnection Architecture

that BIA. This imposes a major reliability impact with multiple BIAs in series to transmit signals along the data bus. Any receiver or transmitter failure of any BIA severely cripples or incapacitates the total data bus and thus system operation.

8.4 DATA BUS PROTOCOL SELECTION

A data-communication protocol is a set of criteria that first establishes and then terminates a connection between processors; identifies the sending and receiving units; accommodates text, programs, control characters, and differentiates among them; and verifies message integrity. A tradeoff analysis was performed to determine the appropriate data bus concept for the electrical power subsystem application. The criteria that were used to select the data bus concept are listed in Table 8-4. The first six criteria are related to cost-effective design practice for a distributed processing digital system. The last two criteria are applicable to the design of distributed processing spacecraft systems for power management and electrical system integration.

Table 8-4. Command and Data Bus Selection Criteria

- Minimum number of signal conductors (or optical fibers)
- Simple hardware implementation (LSI is very desirable)
- Minimum software intervention required to accomplish information transfer
- Data block transfer capability
- Reliable information transfer (error checking capability and simple methods for specifying retransmission of information)
- Widely known and used (desirable for implementing as a standard)
- High bit rate (up to 10 Mb/sec)
- Space qualified hardware (and radiation hardened)

Table 8-5 summarizes the characteristics of six data buses which were investigated. None of these buses satisfy all of the selection criteria; however, the bit-oriented protocols (HDLC, ADCCP, and SDLC) and the MIL-STD-1553B bus are the best candidates. Of these, the MIL-STD-1553B

Table 8-5. Data Bus Characteristics

Command and Data Bus Survey	Basic Features (Bus Structure and Protocol	Software Requirements	Hardware Available	Signal Lines Required	Data Block Capability	Error Checking Capability	Maximum Bit Rate	Remarks
International Standards Organization (ISO) High-level Data Link Control (HDLC); same as American National Standards Institute (ANSI) Advanced Data Communications Control Procedure (ADCCP)	Half-duplex bit-oriented protocol variable length frames	Simple	NMOS LSI (for several micro-processors)	1	Simple block transfers via variable-length data field	16-bit cyclic redundancy Code (CRC) field, and built-in valid frame monitoring	500 kHz to 2 MHz (no standard)	Widely known and used; very flexible command capability; best available for demo breadboard, but available LSI is not rad-hard.
IBM Synchronous Data Link Control (SDLC)	← ESSENTIALLY THE SAME AS ABOVE EXCEPT FOR SOME (DETAILED) LIMITATIONS WHICH REDUCE FLEXIBILITY →							
MIL-STD-15538	Half-duplex Manchester II encoded 17-bit message words	Data management software could be complex	Bipolar LSI	1	32-word maximum	Parity bit per message word	1 MHz	Widely known and used; flexible command capability.
Standard Telemetry and Command Components (STACC) NASA Standard (Goddard)	Full-duplex bi-phase L encoded supervisory (32-bit words) and reply (9-bit words) lines	Data management software could be very complex	No LSI	2	None	Parity bit for supervisory and reply words	1.024 MHz	Widely known; flexible command capability.
General Purpose Interface Bus (GPIB) (IEEE STD 488)	Asynchronous, handshaking type of data transfer with 8 parallel data lines and 8 control lines	Complex	LSI (mostly NMOS, one CMOS)	16	Yes	None	8 MHz over limited distance	Widely known and used; limited to 15 devices per bus.
Flexible Multiplexer-Demultiplexer (FMDM) Sperry	Half-duplex Manchester II encoded 17-bit message words	Complex	No LSI	5	32-word maximum	Parity bit per message word	1 MHz	Flexible command capability; limited to 4 remote modules per central module.

bus was eliminated because of its limited data block capability, complex software, and inferior error checking capability. The bit-oriented protocol buses are essentially the same except for some minor limitations in SDLC. The HDLC was selected because it is general purpose and is widely known and used.

The structure of the HDLC message frame is shown in Figure 8-19 and consists of the start flag, address field, control field, information field, frame checking sequence field, and the end flag. The information field is further subdivided for this application into transmitter address, count, data type, and data fields as shown in Figure 8-20.

The first set of bits to be generated in the format is the start flag field. The configuration of this flag byte is always 01111110, and this sequence will never be repeated again throughout the entire message transmission until the end flag with the same configuration is transmitted. In order to ensure this operation every time, when a sequence of five 1 bits in a row is recognized by the transmit hardware after a start flag is generated (except when the end flag has to be generated), an extra 0 bit will be added to the bit stream between the fifth 1 bit and the next bit, regardless of whether that next bit is 0 or 1. This technique is called "zero stuffing." The hardware at the receive end will always remove a 0 which is detected after five consecutive 1 bits before being passed on to the recognition logic and, therefore, will end up being transparent to the user (Reference 8-2).

The Receiver Address Field is an 8-bit field that identifies the intended receivers of the message. This field provides 127 unique receiver addresses ($2^7 - 1$) and a global address (all other receivers). This utilizes 7 of the 8 bits. The least significant bit (first bit sent) is used to indicate that either another address byte follows (a zero) or this is the final address byte (a one). Multiple, but specific, units may thereby be addressed with a common message. Alternately, the quantity of receivers may be increased beyond 127 by utilizing this address extension technique.

The Control Field is a 16-bit field that typically contains command, response, or sequence numbers. The control field is used by the transmitters to identify special operations for the receiving bus interface

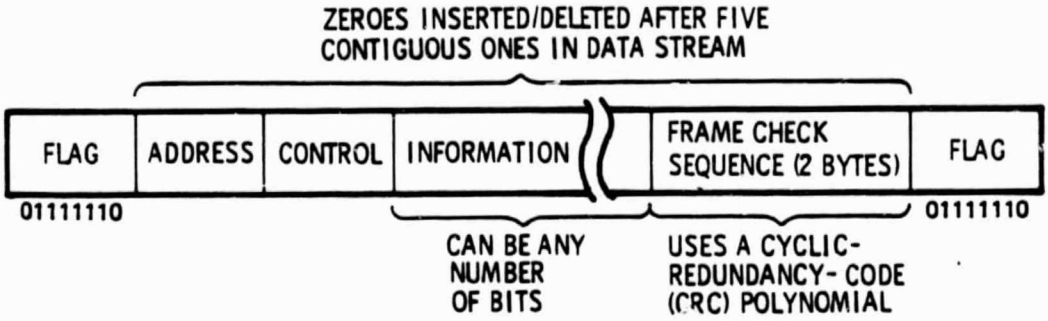


Figure 8-19. ISO (International Standards Organization) HDLC (High-Level Data Link Control) Protocol

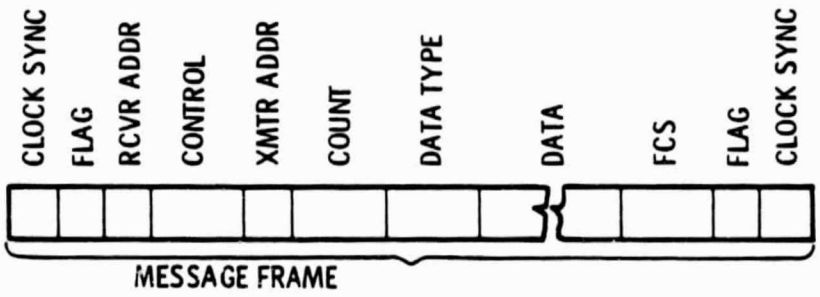


Figure 8-20. Data Bus Communication Format

adapter. Presently, no special operations are identified for electrical power subsystem management, and the field is unused; the field contains 16 zeros.

The Information Field is an even integer multiple of 8-bit groups (bytes) limited by the buffer storage and processing capacity of the bus adapters and processors. Typically, this limit is of the order of 1000, 2000, or 4000 bytes! (Two thousand is provided in the electrical power subsystem bus adapter design.) This information field is subdivided into transmitter address, count, data type, and data fields for application to electrical power subsystem control.

The Transmitter Address field is an 8-bit field that identifies the transmitter of each message frame. This address field provides 127 unique transmitter addresses (the same units as the 127 receiver addresses). The global address is not used. The least significant bit (first bit sent) is used to indicate address field completion, i.e., a zero means more 8-bit fields to follow, a 1 indicates this is the last 8-bit address field. The quantity of transmitters may thereby be increased beyond 127 by utilizing this extension technique.

The Count Field is a 16-bit field indicating the total length in bytes of the following two fields: data type and data.

The Data Type field is a 16-bit field indicating the type of data in the message: command, equipment response, telemetry data, equipment health, status monitor data, etc.

The Data Field is an even integer multiple of 8-bit groups (bytes) limited by the buffer and processor capacity of the bus adapters and control processors. This field contains the coded data message from the transmitter to the receiver(s).

The Frame Checking Sequence (FCS) is a 16-bit field which is an error checking code generated from the bits in the message frame, excluding the start and end flags and the FCS field (Reference 8-3). This ensures the received message can be judged to be error free in transmission and reception. Erroneous data is so noted, and software accommodates request(s) for retransmission.

A Flag Field precedes and follows each message block. The flag sequence, serial "01111110," must be unique. Hence, the transmitting bus interface adapter inserts a zero bit in the transmitted data bus bit sequence after each fifth consecutive one bit of all other fields, including the frame checking sequence. The receivers, in turn, automatically discard the zero after five contiguous one bits, thereby decoding the message block to the intended bit pattern. Six contiguous one bits thereby remain unique as the flag.

A Clock Synchronization Field (clock sync) precedes and follows each frame. The clock is an 8-bit field of alternating zeros and ones; "01 01 01 01" (serial).* This provides a series of clocking pulses to allow the Manchester decoder in the receiver to synchronize with the transmitter encoder bit interval before data transmission commences. The trailing field is required to maintain synchronization during the processing period (several bit times) following the end-of-message flag.

8.5 DATA BUS CONTENTION RESOLUTION**

At the conclusion of a message over the data bus, a contention problem exists to determine which data bus interface adapter (BIA) is allowed to transmit next over the data bus. A clear resolution of this contention for the data bus is mandatory, or two or more units (BIAs) may elect to transmit simultaneously resulting in garbled and unintelligible messages. The selected contention scheme was derived from several candidates and operational options to assure communication capability even with some equipment failures. This contention scheme is designed to complement a global data bus architecture.

---

*Two-bit pattern representations are employed in digital engineering for parallel and serial bit display. Parallel display produces a mathematical bit pattern with the most significant digit to the left: 2^n, ..., 2^3, 2^2, 2^1, 2^0. This is the video (CRT) display of processor logic registers (parallel bit transfer). Serial display produces a bit pattern with the first bit sent written to the left, typically with the least significant digit first: 2^0, 2^1, 2^2, 2^3, ..., 2^n. This is the oscilloscope display of serial data transfer. Discipline must be exercised to define which pattern a digital expression represents as two interpretations are possible without clarifying commentary (see Appendix D).

**The data bus contention resolution was developed on the TRW Internal Research and Development Program "Advanced Remote Integration Assembly," 1981.

The contention for transmission on the data bus is resolved by utilizing a distributed-controlled, time-sequential access scheme. Each BIA is allowed to transmit on the data bus in consecutive order; the order of transmission follows the same sequence as the ascending order of BIA addresses. When a BIA determines that it has access to the data bus, it must transmit at least one frame; if there is no information frame to be transmitted, the BIA will output a "token" frame, which is addressed to all other BIAs on the bus (using the global, all ones, address). A token frame, with preceding and succeeding CLOCK SYNC bytes, is illustrated in Figure 8-21.

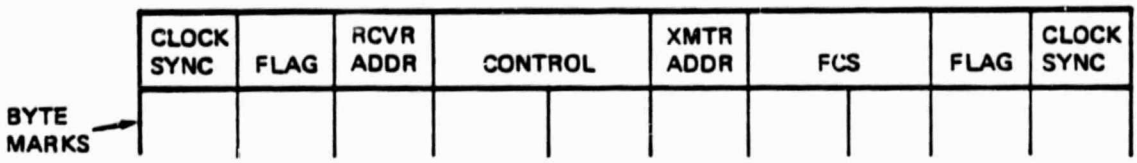


Figure 8-21. Token Frame

If the controller processor has presented its associated BIA with an information frame to be transmitted when access to the data bus is gained, this information frame is transmitted first. Each BIA is allowed to transmit only one information frame (a maximum of 2050 bytes) per bus access period. A token frame is then transmitted, with the closing flag of the information frame serving also as the opening flag of the token frame, as illustrated in Figure 8-22.

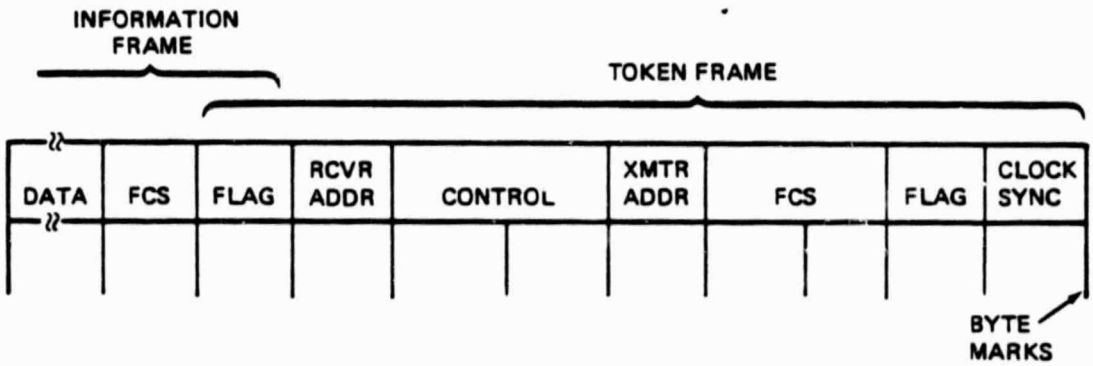


Figure 8-22. Message Frame and Token Frame

When a BIA receives a token frame, it will compare its address to the address of the BIA that sent the token frame (XMTR ADDR field). Each BIA which receives a token frame will then calculate the maximum number of BIAs which could gain bus access prior to its next access period; for convenience this number is called the access priority number. To allow for the (reasonably probable) case where one or more BIAs with lower access priority numbers are not able (or on-line) to transmit token frames, each BIA will decrement its access priority number by one after a delay of 14 byte times (112 bit intervals on the serial data bus) if no FLAG is observed on the data bus. This 14-byte delay time was derived from three components, two byte times to receive the CLOCK SYNC and FLAG fields, 10 byte times to allow for the case when any BIA receives an erroneous frame, and two byte times for the maximum round-trip propagation delay of the data bus. Justification for a maximum one-way propagation delay of one byte time is given in Appendix F. If a FLAG is still not observed on the data bus after the first decrementing of the access priority number, each BIA will continue to decrement this number by one for every two byte times of delay, allowing only for additional maximum round trip propagation delays on the bus. Once a FLAG is observed on the data bus, each BIA will update its access priority number after receipt of the forthcoming token frame. If an erroneous frame (either information or token) is received, each BIA will delay for ten byte times and then decrement its access priority number by one. Once a BIA's access priority number reaches zero, it will have gained access to transmit on the data bus.

When system power is turned on, all BIAs are reset to the IDLE mode. Under software control, the system controller BIA will enter the TRANSMIT-TOKEN mode, and start to send token frames. Other BIAs in the system will then be placed in the RECEIVE mode via software control. This power-up sequence will properly initialize the data bus contention resolution scheme.

A second data bus contention scheme that was considered is carrier sense multiple access/collision detect. In this scheme, if the controller has data to send, it first listens to the data bus to see if any other unit is transmitting. If the data bus is busy, the controller waits until it becomes idle if the controller has data to send. When the controller

detects an idle data bus, it transmits a message. If, by chance, two controllers try to transmit simultaneously, a collision occurs. If a collision occurs, the controller waits a random amount of time and starts over again (Reference 8-4). The carrier-sense, mulitple-access scheme provides the simplest hardware implementation but was rejected because of the complicated statistically based software required to respond to data bus collisions.

The third data bus contention scheme that was considered is called a "bit-map" method. In this scheme, a contention period is allotted to the data bus. The contention period contains one time slot for each controller on the data bus, as shown in Figure 8-23. If Station 0 has a message to send, it transmits a one bit during the first slot. No other controller is allowed to transmit during this slot. Regardless of what Station 0 does, Station 1 gets the opportunity to transmit a one during Slot 1 if it has a message to send. In general, Controller j may announce the fact that it has a message to send by inserting a 1 bit into Slot j. After N slots have passed by, each controller has complete knowledge of which controllers wish to transmit. At that point they begin transmitting in numerical order. Since everyone agrees who will go next, there will never be any collisions. After the last ready controller has transmitted his message, an event all controllers can easily monitor, another N bit contention period is begun (Reference 8-4).

The bit-map method was rejected because it requires complicated hardware and a central clock for continuous synchronization; thus it is vulnerable to single-point failures.

8.6 MICROPROCESSOR SELECTION

Commercially available microprocessor candidates were identified and evaluated for application as power source, load center, and electrical power subsystem controllers in a distributed power management subsystem. The principal functions required of the microprocessor controllers are power switching decisions, command and data handling, and command decoding. Only moderate arithmetic processing capability is needed for battery charge control and system load balancing algorithms. Hence, there are many microprocessors (Table 8-6) that will meet the functional requirements for these

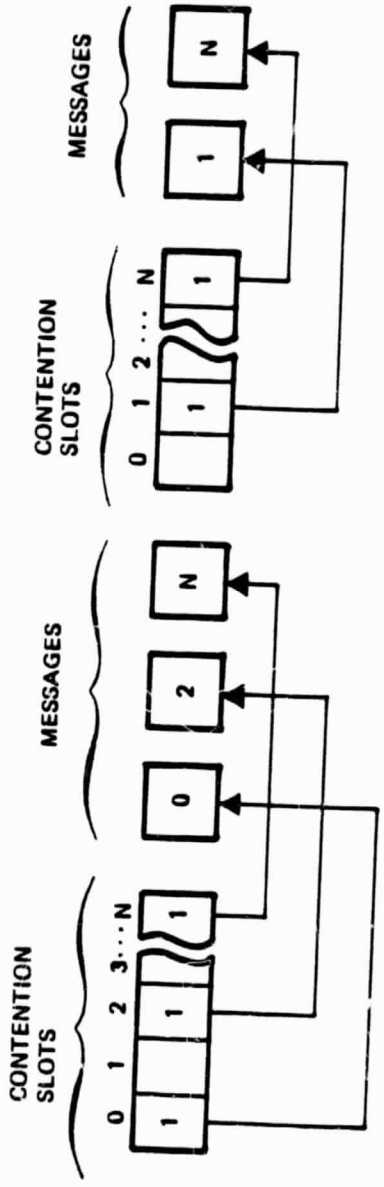


Figure 8-23. The Basic Bit-Map Method

Table 8-6. Comparison of Candidate Microprocessors

Microprocessor	Word Length	Technology	Advantages	Disadvantages	Remarks
1. SBP9900 (TI)	16 bits	I^2L	Qual'fied; radiation hard. Fits Future Data and Tektronix 8002 development systems.	I^2L devices have higher dissipation than CMOS.	Memory-to-memory architecture is significantly different from other microprocessor families.
2. MD46802 (Mitel)	8 bits	ISO - CMOS	Fastest of candidate microprocessors; very capable architecture; low power dissipation	New device (early 1981 availability); not space qualified.	MC6800 architecture and instruction set; fits Tektronix 8002 development system.
3. NSC800 (National)	8 bits	CMOS	Capable architecture and I/O devices; low power dissipation.	New device (early 1981 availability); not space qualified. Does not fit Tektronix 8002 development system.	Z80 architecture and instruction set.
4. 80048 (Intersil)	8 bits	CMOS	Single-chip computer architecture; low power dissipation.	New device (early 1981 availability); not space qualified. Limited DMA capability.	8080 architecture and instruction set.
5. SBP9989 (TI)	16 bits	I^2L	Additional arithmetic capability over the SBP9900; radiation hard.	New device (1981 availability); not space qualified.	Memory-to-memory architecture.
6. MC146805 (Motorola)	8 bits	CMOS	Single-chip computer architectur ; low power dissipation.	New device (early 1981 availability); not space qualified. Limited DMA capability.	MC6800 architecture and instruction set.
7. 8086 (Harris)	16 bits	CMOS	Capable architecture and I/O devices; low power dissipation.	Not available until 1982 (or later).	Will fit both Future Data and Tektronix 8002 development systems in 1981.

Table 8-6. Comparison of Candidate Microprocessors (Continued)

Microprocessor	Word Length	Technology	Advantages	Disadvantages	Remarks
8. CDP1802 (RCA)	8 bits	CMOS	Low power dissipation.	Lowest capability (of candidates) instruction set and architecture; having difficulty achieving space qualification.	Fits Tektronix 8002 Development System.
9. IM6100 (Intersil)	12 bits	CMOS	Lower power dissipation; good software support; capable architecture.	Does not fit Tektronix 8002 nor Future Data development systems; not space qualified; additional hardware required to expand address space beyond 4096 words.	Recognized PDP-8 instruction set.

controller applications. Selection is therefore made on other factors which relate to a sound system design, such as:

a) Availability

b) Widely used, with good software and hardware design support

c) Extensive family of input/output devices

d) Capable architecture, with adequate speed and instruction set

e) Space qualified, or radiation hardenable

f) Compatible with Tektronix 8002A and Future Data development systems

g) High level language capability

h) Low (or moderate) power dissipation.

Another system design goal is to provide enough flexibility so that upgrading to a more powerful microprocessor is possible without complete hardware and software redesign. This upgrade mobility permits an increase in processing capability as the application of microprocessor control expands with design experience and becomes more sophisticated. Thus, an important selection guideline is that the microprocessor be one of an evolving family of microprocessors with compatible input/output characteristics and upward compatible software (where possible). This permits a comparatively inexpensive step increase in system processing capability. This upgrade mobility is identified (Figure 8-24) for the various microprocessor candidates.

Nine candidate microprocessors were identified and compared (Table 8-6). The Texas Instruments SBP9900 microprocessor is the only one which satisfactorily meets all the selection criteria. The recent announcement of a new Texas Instruments microprocessor (available in 1981), called "Alpha", brings the 9900 family in line with the guideline related to future system enhancement by upgrading with a more capable microprocessor. The TI "Alpha" will have capabilities to make it competitive with the Z8000 (Zilog, Mostek, AMD) and the MC68000 (Motorola), and its instruction set will be upward compatible with the 9900 and the 9989 microprocessors. Although the MC68000 and the Z8000 are perhaps the most powerful (16-bit) microprocessors on the market, they are NMOS devices (radiation soft).

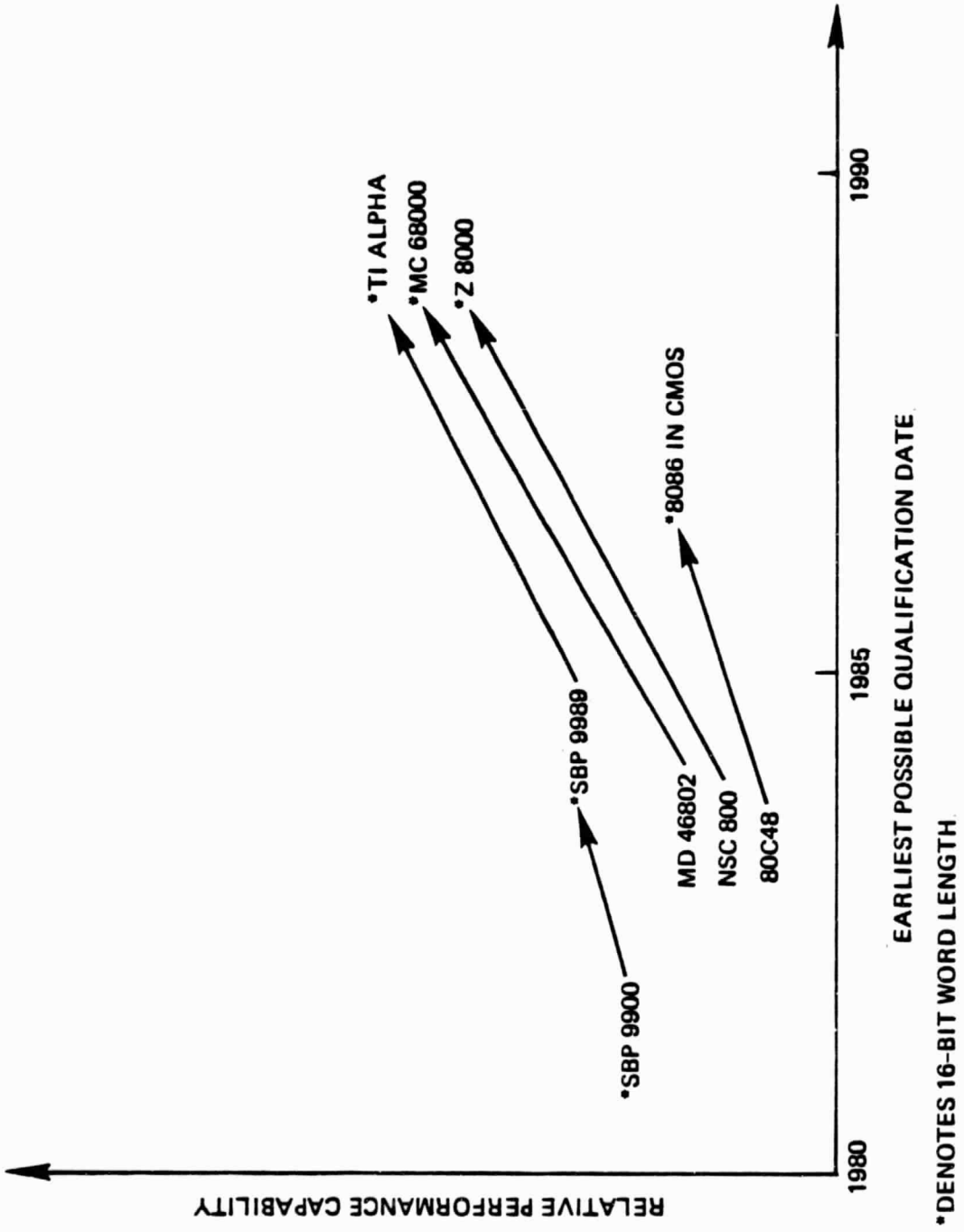


Figure 8-24. Possible System Upgrading Stages

Also, there are no CMOS versions of these two devices in the foreseeable future. Hence, the significant advantages of the TI SBP9900 outweigh the negative factor of the "unknown quantity" in the Alpha enhancement. Therefore, the TI SBP9900 is selected for the power management subsystem controllers primarily because a capable space-qualified device is available and presently available development systems can be utilized.

8.7 SELECTION OF FORTH AS THE HIGH LEVEL LANGUAGE FOR DISTRIBUTED-PROCESSING ELECTRICAL POWER SYSTEMS

The decision to use a high-level language to program the distributed-processing within the Electrical Power System was made primarily to reduce the cost and time required to develop software entirely in assembly language. Since high level languages are usually quicker to learn than assembly languages, less time should be required to develop applications programs. A second requirement was the selection of a language which facilitates structured design, which would allow the development of modular software. This property will decrease development time and increase the testability and maintainability of the system.

Hardware limitations also restrict the selection of a high level language. Although the selected microprocessor (the SBP9900 from Texas Instruments) will support several high level languages, all except FORTH and TI's Power Basic would require the purchase of a development system. One of the strongest advantages of FORTH is its ability to be a complete stand-alone development system in less than 6K bytes of memory in the target system. Since Basic does not facilitate structured programming, FORTH is clearly the best choice from a hardware cost point of view.

The requirement for a high level language which supports structured design, coupled with the strong desire for a language with fairly wide-spread usage, has restricted the choice to Ada, PL/M, Pascal, and FORTH. Since Ada compilers are not yet available, detailed investigations were limited to FORTH, PL/M, and Pascal. Each of these languages has its strong and weak features; i.e., no language is best for all applications. The evaluation and selection process should be tailored to fit the specific application, a real time spacecraft control system based upon functionally distributed processors.

FORTH has properties which fit the application remarkably well. A fundamental feature of FORTH is that the language and operating system reside in the same entity, a small set of constructs (called "words") which allow the user to build any new constructs, data structures, etc., that fit his particular application. Thus, FORTH has the property of extensibility (expandability and flexibility) which greatly facilitates incremental development of software in small steps. And FORTH explicitly contains the control constructs required for structured programming.

The fact that FORTH provides the language, operating system, and stand-alone development system (in less than 6K bytes of memory) for each processing node is significant to the application. This feature will allow interactive dynamic testing of the distributed-processing system when multiple processors are actively operating in the system. This will be a very useful feature for test and evaluation of data bus protocols, and it will allow thorough interactive dynamic system testing to be a realizable goal at a reasonable cost. The current state-of-the-art test equipment for distributed-processing systems simply does not have comparable capability at a reasonable cost; a complete development system for each processing node in the system under test would be required to provide such capability.

Execution time of code written in FORTH can approach 50 to 70 percent of the speed of assembly language code, and the FORTH code occupies less memory (60 to 80 percent of the equivalent assembly language code). Programs written in Pascal or PL/M use approximately twice as much memory space as FORTH code, and execution times of Pascal or PL/M programs are significantly slower than programs written in FORTH. A report issued from General Electric Ground Systems Department has recommended the selection of FORTH as the standard language for all embedded processors in a distrubuted-processing image generation system (Reference 8-5). Table 8-7, which is a simplified version of a table in the report of Reference 8-5, shows the relative evaluation (on a scale of 1 to 10, where 10 is "best") of several factors involved in selecting high level languages. The author of the report claims that, "The data was secured from interviews with about two dozen expert programmers from business, scientific, and real-time backgrounds."

Table 8-7. Comparison of High Level Languages

Factors	PASCAL	FORTRAN	BASIC	PL/M	COBOL	FORTH
1. Applicability for Large Programs	10	6	3	8	7	10
2. Applicability for Small Programs	8	7	10	8	3	10
3. Ease of Learning	6	8	10	7	4	6
4. Supports Top-Down Design	10	5	3	7	4	10
5. Control Constructs Supporting Structured Programming	10	4	4	10	6	10
6. Writeability of Large Programs	10	5	3	9	5	10
7. Capable of Generating Efficient Code	8	7	6	10	7	10
8. Flexibility of Data Types	7	5	1	6	7	10
9. Capability of Compiler to Check for Consistency and Errors	10	5	4	6	7	7
10. Controlled Access to Data	10	6	1	3	4	10
11. Language Supports Effective Problem Solving	10	6	1	8	7	10
12. Readability of Large Programs	10	4	2	8	8	10
13. Language Supports Transportability	7	7	4	2	9	10
14. Effectiveness of Standards	3	10	4	3	10	9
Totals*	119	85	56	95	88	132
Average Evaluation	8.5	6.1	4.0	6.8	6.3	9.4

*Each factor given an equal weight (=1).

The relative evaluations of compiler checking capability (Item 9, Table 8-7), the capability of the language compiler to check for consistency and errors (in data operations, syntax, etc.), shows a significant weakness of FORTH, especially when compared to Pascal. For example, Pascal has (and Ada will have) the ability to locate inconsistencies in manipulating data; implicit mixed mode operations on data are prohibited. FORTH allows mixed mode operations, which causes some errors to be harder to detect. FORTH will also allow the definition of an array subscript which is beyond the limits of the array definition. Another disadvantage of FORTH, when compared to Pascal or Ada, is that FORTH tends to encourage an undisciplined programmer to use complex "tricks" instead of simpler code which is more easily read and understood. In order to help alleviate the disadvantages mentioned above, a set of programming rules and procedures will be defined before any FORTH programs are written. With the aid of an experienced FORTH programmer, a realistic discipline for FORTH programming will be established.

Another commonly voiced "criticism" of the FORTH language is that its stack architecture and its associated notation result in programs which are not easy to read. For passing parameters and returning results, the FORTH programmer uses a data stack, which he operates via reverse-Polish-notation (RPN).

Some users find RPN expressions difficult to read, but experience has shown that RPN is easily learned even by elementary students, not to mention the users of many calculators. Why a data stack and RPN instead of the more traditional algebraic expression? A stack architecture offers several advantages (Reference 8-6):

- Parameter passing between modules becomes implicit and extremely simple.
- Problems associated with initialization and the validity range of variables in blocks disappear. Local variables, kept on the stack, need no names.
- Variables are transparent to the user - a variable is simply an address and the content of this address is a value. The availability of addresses allows complex pointer calculations.
- Module testing is very simple. The user explicitly puts parameters on the stack, in the interactive mode, to debug the module at hand.

- Words are executed in the order in which they are given to the system, making execution flow explicit. The programmer does not have to outguess the compiler.

In summary, FORTH is highly rated in twelve of the fourteen factors in the enclosed comparison table, and it rates highest overall of the six widely used high level languages. In additon, FORTH has decided advantages in execution speed, memory compactness, and significant advantages for system testing considerations, both economic and technical. It is clear that FORTH is the best available high level language for the application.

9. CONCLUSIONS AND RECOMMENDATIONS

A power management system concept has been presented for a 250-kilowatt utility-type electrical power subsystem for a low-earth orbit, large-space platform. This concept utilizes a decentralized data processing approach to achieve autonomous operation of the electrical power subsystem. The algorithms required to perform the power management functions were identified and algorithm development was initiated with the development of load bus assignment, command processing, and monitoring algorithms. The power management system hardware development was initiated with the assembly of an electrical power subsystem controller and a load center controller.

9.1 CONCLUSIONS

The major conclusions concerning the power management system are as follows:

a) The power management system concept (as presented herein with autonomous operation) is a cost effective approach from the standpoint that it reduces ground station operational costs, reduces shuttle resupply costs, and extends equipment life.

b) An on-board power management system, as presented herein, will be required (enabling technology) for large space platforms due to increased data requirements imposed by system complexity and component interactions. This conclusion assumes that present telemetry channel constraints will continue to exist. The increased complexity of the electrical power subsystem will require computerized analysis of system anomalies and degraded components in order to provide corrective action within an acceptable time frame.

c) Power management provides an alternate path to technology readiness of large space platform power subsystems, rather than having to develop large power subsystem components, by providing the capability to operate more complex systems.

9.2 TECHNOLOGY DEVELOPMENT RECOMMENDATIONS

Technology developments for the power management system fall into two major categories: hardware development, and algorithm development. State-of-the-art developments in power management hardware are such that only one major development appears to be required. This is the development of light weight, low power, accurate sensors which include current and pressure

sensors. The large number of sensors required to control and understand the status of the electrical power subsystem is the major driver towards this development.

Other hardware developments, such as microprocessors, appear to be progressing rapidly due to data processing requirements in the commercial sector. Although only one radiation hardened microprocessor is available at this time, several capable CMOS processors are coming on the market that appear adequate for the power management system applications. Data communications technology such as data bus protocols are being developed by the commercial sector for distributed processing applications in banks, airlines, etc., that appear to be more than adequate for the power management system application.

The major technology development that is required is the development of the electrical power subsystem algorithms (operational strategies/ procedures). The autonomous operation requires preplanned decisions to be made and implemented into the on-board controllers. The utility-type operation requires a new approach to power management in that load profiles are unknown and redundancy management is required with degraded operation for full system utilization. Shuttle resupply requires new operational techniques in that degraded performance must be analyzed and predicted in order to minimize the number of shuttle resupply trips that are required.

10. REFERENCES

5-1 Tomlinson G. Rauscher, "A Unified Approach to Microcomputer software development" Computer, June 1978.

8-1 Bob Stevens, Division Manager, Telecommunications Division of Jet Propulsion Laboratory, personal conversation.

8-2 Kenneth Sherman, Data Communications, Reston Publishing, Co., 1981.

8-3 ISO 3309, Paragraph 3.6, Frame Checking Sequence (FCS).

8-4 Andrew S. Tanenbaum, Computer Networks, Prentice-Hall, 1981, p. 296f.

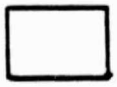
8-5 R. Johnston, "Programming Language Study Report," General Electric Ground Systems Department, 4 June 1979.

8-6 Michel Mannoni, "FORTH - an Extensible Path to Efficient Programs," Electronic Design Magazine, 19 July 1980.

APPENDIX A

LEIGHTON DIAGRAM SYMBOL DEFINITION

The structured design of an algorithm is typically illustrated by a Leighton diagram. The diagram illustrates processes, decisions, sequence of events, and interfaces. The sequence of events is defined by reading the diagram from top to bottom and left to right. The symbols in the diagram are defined below:

a)  A box indicates a process.

b)  A diamond indicates a decision.

c) An arrow indicates that the processes are to be performed in sequence or in a loop.

d)  A parallelogram indicates an input or output destination.

e)  A shaded box indicates that this process is for demonstration only, and would not be included in a flight program.

APPENDIX B

TI TM-990 MODULE DESCRITPIONS

Engineering demonstration consoles were assembled for the Electrical Power Subsystem Controller (EPSC) and the Load Center Controller (LCC). These consoles utilize the Texas Instruments TM-990 series microcomputer modules for the microprocesssor, memory, interface (I/O) circuitry, and mounting hardware. The following sections provide a brief description of the functions and features of the selected hardware. For more complete detailed descriptions, refer to the TM 990 Microcomputer Catalog (handbook) by Texas Instruments.

1. TM 990/101MA-3 MICROCOMPUTER CARD

The TM 990/101MA-3 is a preassembed and tested microcomputer module (Figure B-1) which utilizes the 16-bit, NMOS, TMS 9900 microprocessor as its central processing unit. RAM and ROM/EPROM memory and programmable serial and parallel data inputs/outputs are included on the card assembly. Major features are:

- TMS 9900 16-bit CPU
- 4K bytes of TMS 2114 static RAM
- 2K bytes of ROM
- 16 programmable I/O lines
- Two RS-232C serial ports
- Fully expandable bus structure and DMA to on-board memory
- 3 MHz operation
- Three programmable interval timers
- Seventeen interrupts: 2 nonmaskable, and 15 prioritized and maskable
- Edge triggered interrrupt, with software reset
- Communications Register Unit (CRU) addressable LEDs and DIP switches for custom applications
- Designed to fit TM 990/530 card cage.

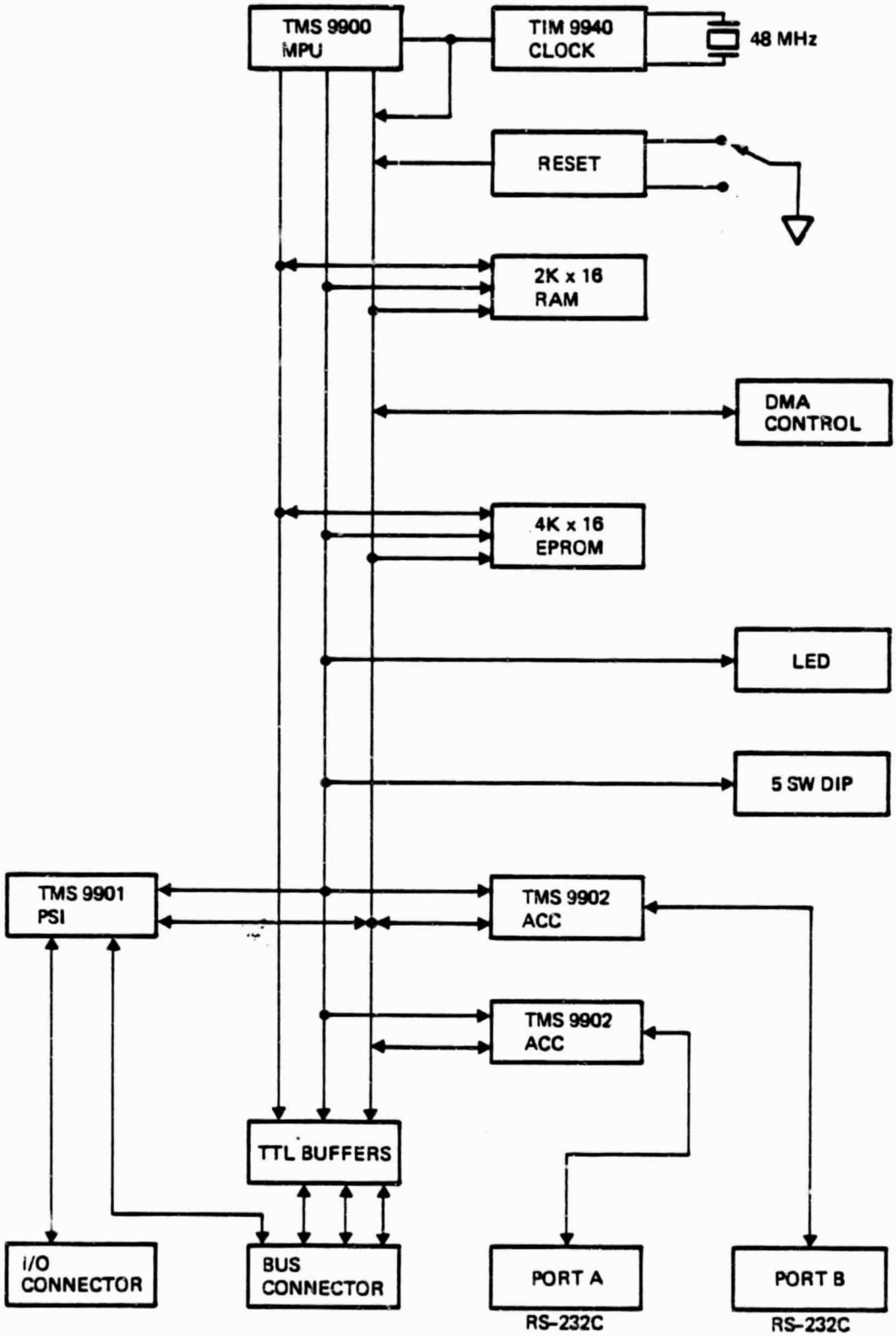


Figure B-1. TM 990/101MA-3 Block Diagram

2. TM 990/201-42 MEMORY CARD

The TM 990/201-42 is a preassembled and tested memory expansion board designed for use with the TMS-9900 based microcomputer modules such as the TM 990/101MA-3. The TM 990/201-42 is socketed for both static RAM and EPROM memory and is expandable to a maximum configuration of 16K bytes of RAM and 32K bytes of EPROM. Major features are:

- Bus compatible with the TM 990/101MA-3 microcomputer card
- 16K bytes of TMS 2716 EPROM installed; expandable to 32K bytes
- 8K bytes of TMS 2114 static RAM installed; expandable to 16K bytes
- Memory map is completely configured
- 1 microsecond cycle time (3 MHz)
- TTL-compatible interface
- Designed to fit the TM 990/530 card cage.

3. TM 990/310 DIGITAL I/O CARD

The TM 990/310 is a preassembled and tested input/output expansion card (Figure B-2). This card provides expansion of 48 I/O points that are programmable as either input or output. Major features are:

- Compatible with the TM 990 microcomputer module CRU bus
- Up to 27 I/O lines may be programmed as prioritized, unlatched interrupts
- Three (+) and three (-) edge-triggered and latched, prioritized interrupt inputs are provided (in addition to 48 I/O lines)
- Contains three real-time clocks (or event timers)
- I/O lines are provided with echo-back feature.
- Inputs/outputs are TTL-compatible
- May be used with solder, wire wrap, or ribbon cable edge connectors
- Designed to fit the TM 990/530 card cage.

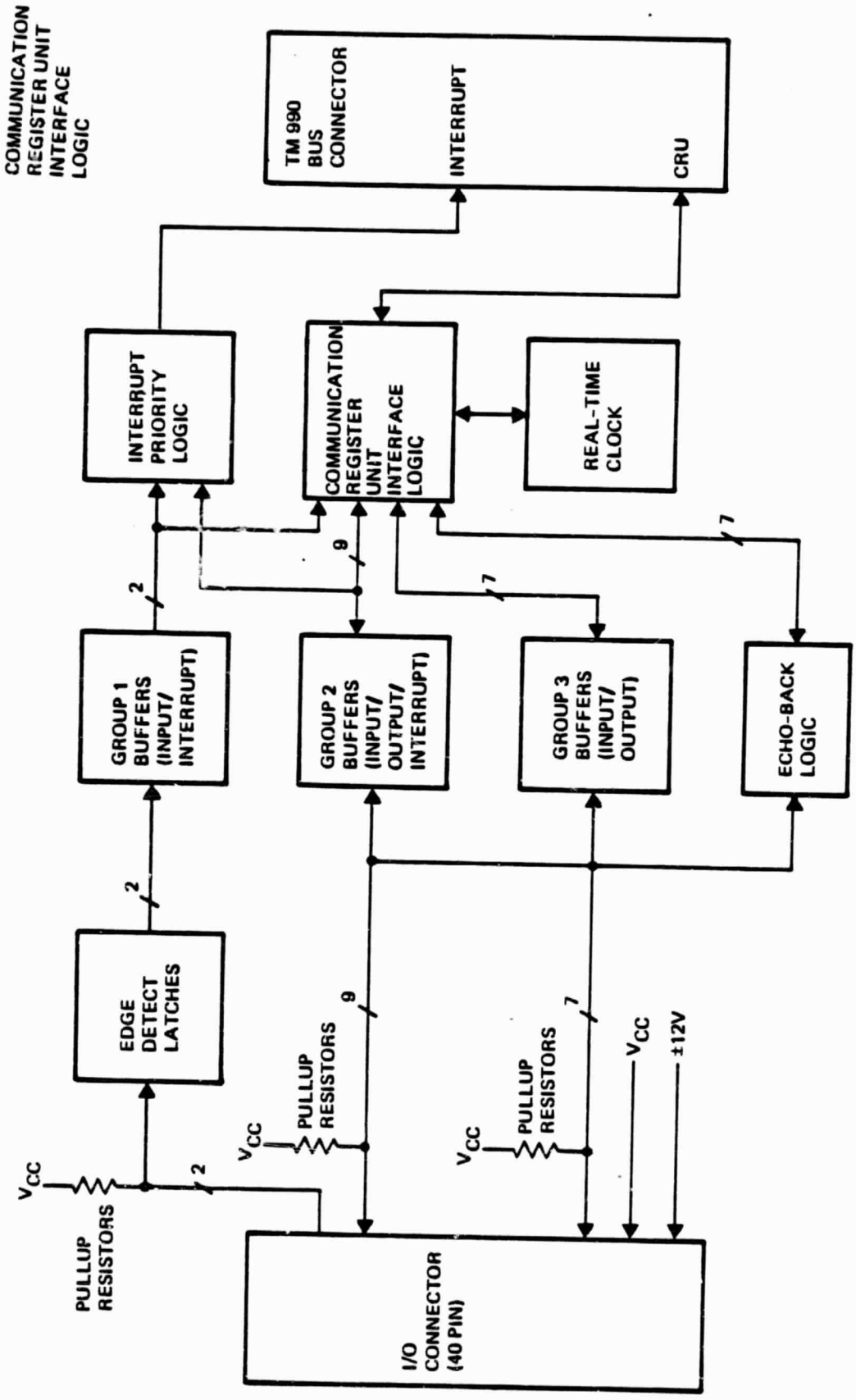


Figure B-2. TM 990/310 Block Diagram

4. TM 990/1240-R ANALOG I/O CARD

The TM 990/1240-R is a preassembled and tested analog I/O card. This card converts analog input signals to a 12-bit digital form (Figure B-3). A multiplexer is included to intermix 16 single-ended analog signals for conversion. The data bus, control bus, and address bus connections to the interfacing microprocessor are included on the card cage connector wiring (TM 990/530). Major features are:

- Memory mapped I/O interface
- 12-bit resolution and accuracy
- Single +5 volts power
- 256 input channel expansion capability
- Input overvoltage protection
- Resistor programmable input gain
- Interrupt operation capability
- Designed to fit TM 990/530 card cage.

5. TM 990/5MT43 DISCRETE INTERFACE MOUNTING BASE

The TM990/5MT43 interface assembly is designed to handle ac and dc "input" functions. The interface assembly consists of the TM 990/5MT43 mounting base and plug-in interface modules (Table B-1). Both the input and output modules are solid state and are optically isolated. Internal protection against harmful voltage transients and RFI noise is provided in the system to eliminate the need for external filters clippers, or suppressors. The mounting base will accommodate up to 16 modules.

Major features are:

- Designed for industrial applications
- Individual plug-in modules
- Prewired mounting base
- Optical coupled isolation
- One I/O point per module

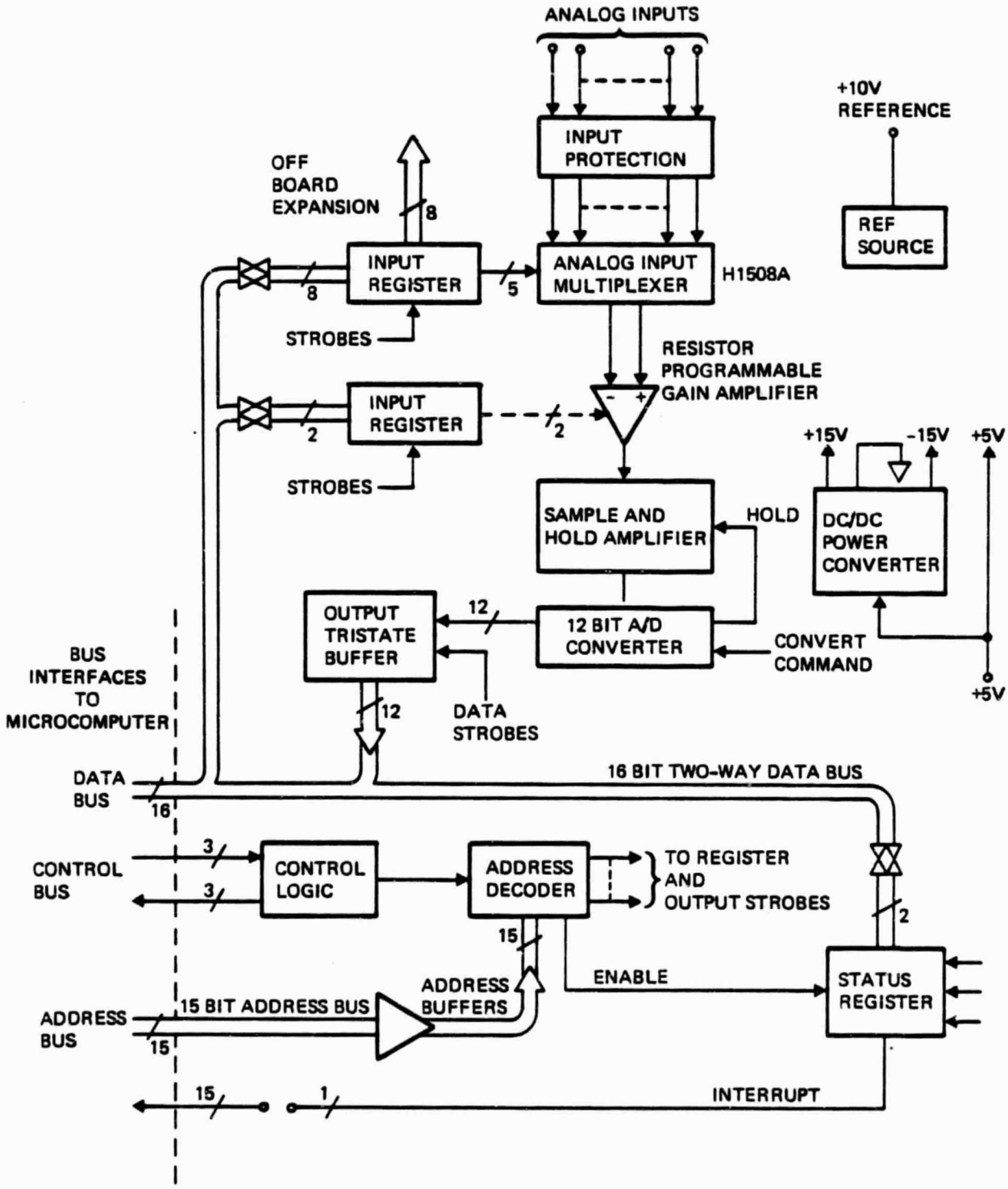


Figure B-3. TM 990/1240-R Block Diagram

- LED status indicator
- Input and output modules
- Compatible with TM 990 microcomputer system
- AC, DC modules
- 1500-volt isolation between input and output.

Table B-1. Module Availability

Catalog No.	Type of Device	Rating	
		Voltage	Current
TM 990/5MT1A05L	AC Input	90-132 VAC	35 mA Max
TM 990/5MT1E05L	AC Input	17-28 VAC	18 mA Max
TM 990/5MT240AL	AC Output	90-132 VAC	3 A Max
TM 990/5MT240EL	AC Output	17-28 VAC	3 A Max
TM 990/5MT3D03L	DC Input	3-28 VDC	30 mA Max
TM 990/5MT430CL	DC Output	10-28 VDC	1 A

6. TM 990/303A FLOPPY DISK CONTROLLER CARD

The TM 990/303A is a preassembled and tested control card designed to interface between the TM 990/101MA-3 microcomputer card and selected disk drives (Figure B-4).

- Shuggart model SA 400 (mini)
- Shuggart model SA 800 (standard)
- Control data model CDC 9404B (standard)
- Qume DT[illegible] (standard double-sided).

The TM 990/303A also provides DMA capability with the TM 990/101MA-3 microcomputer board or with any expansion memory board (e.g., TM 990/201-42). Major features are:

- Formats supported (soft-sectored):
  - IBM single density
  - IBM double density

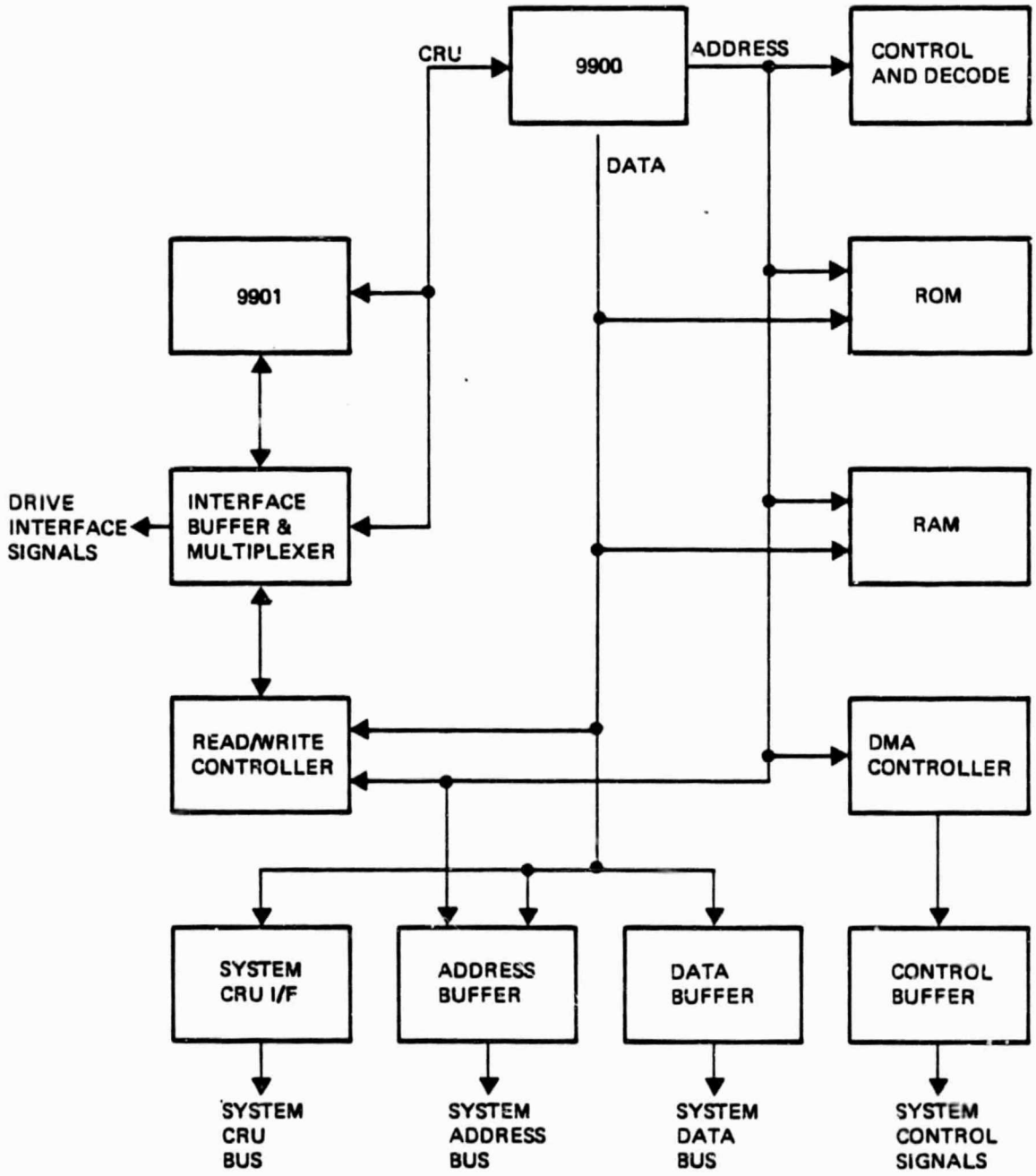


Figure B-4. TM 990/303A Block Diagram

  - TI single density FD800 (currently used on the FS 990/4 and AMPL systems
  - TI Digital Systems Group (DSG) double density format
- Disk sizes: Standard or mini
- Disk sides: Single or double (with Qume DT/8)
- Number of disk drives (daisy chained): Four maximum standard size or three maximum for mini size
- Recording methods:
  - single density frequency modulation (FM)
  - double density modified frequency modulation (MFM)
- Data structure:
  - TF FS 990 compatible
  - IBM 3740 compatible
- System interface:
  - CRU (controller initialization)
  - DMA transfer (data and commands)
- Three LEDs indicate controller status
- Bootstrap load features can be used to initialize system from diskette
- Controller firmware provided on two TMS 2716s (2K words); controller firmware EPROM space expandable to 4K words by using two TMS 2532s
- DMA data transfer
- 20-bit host memory addressing
- Read after write
- Disk command chaining.

Software on the controller includes the following features:

- Seventeen commands including controller self test, read and write to/from diskette, read to and write from controller/RAM, bootstrap load from diskette software, format diskette, execute program in controller memory, and read status of specified drive

- Command completion interrupt to host (interrupt level jumper selectable); completion status reported to host
- Controller initialization through interrupt via CRU

7. TM 990/530 CHASSIS

The TM 990/530 is a preassembled chassis designed for 19-inch rack mounting (Figure B-5) that accommodates 16 cards and provides backplane interconnections for the address bus, data bus, interrupt, and control lines. The TM 990/530 chassis can accomodate the addition of cooling fans (Figure B-6) for forced-air cooling of chassis mounted cards.

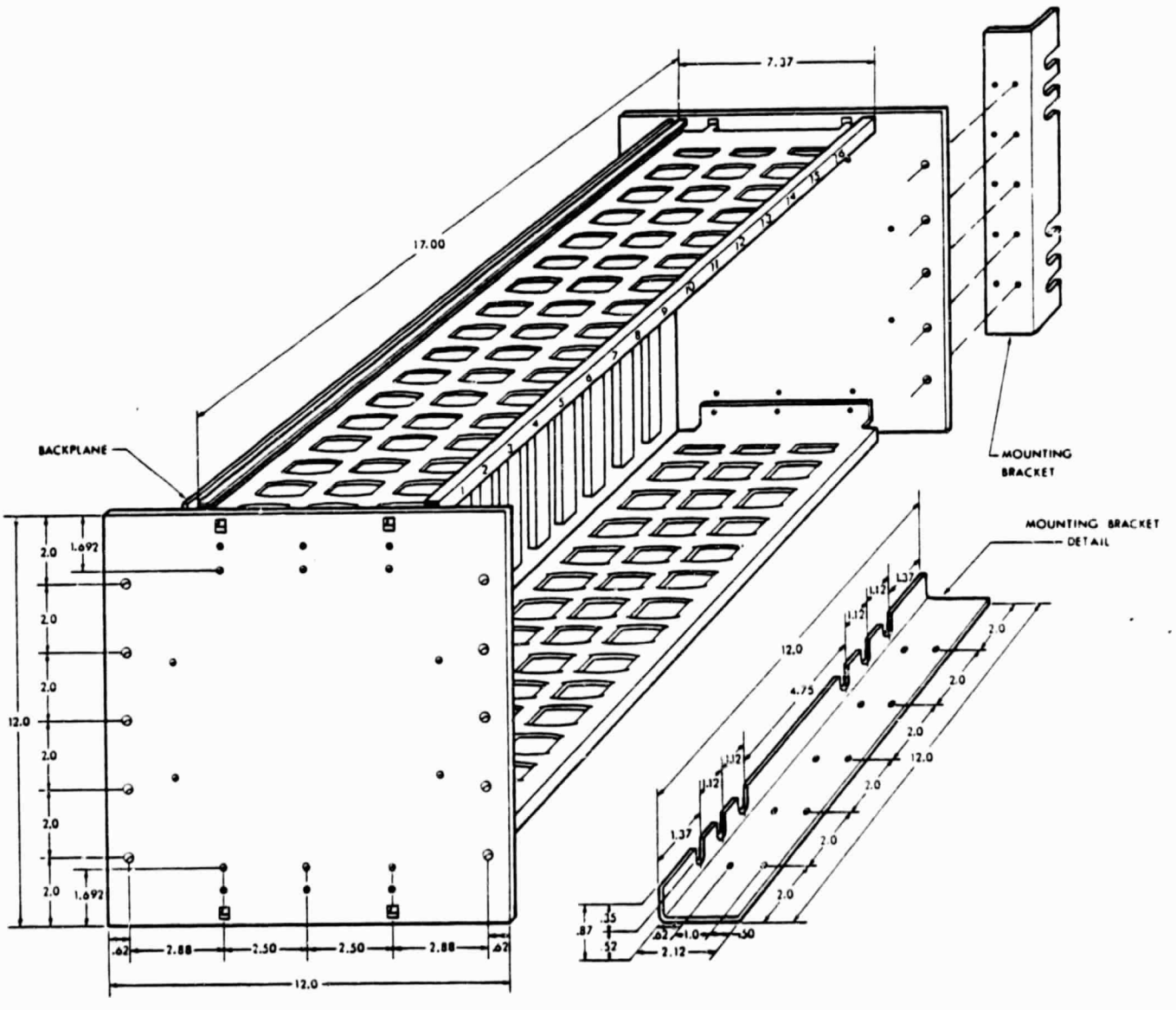


Figure B-5. TM 990/530 Chassis Dimensions

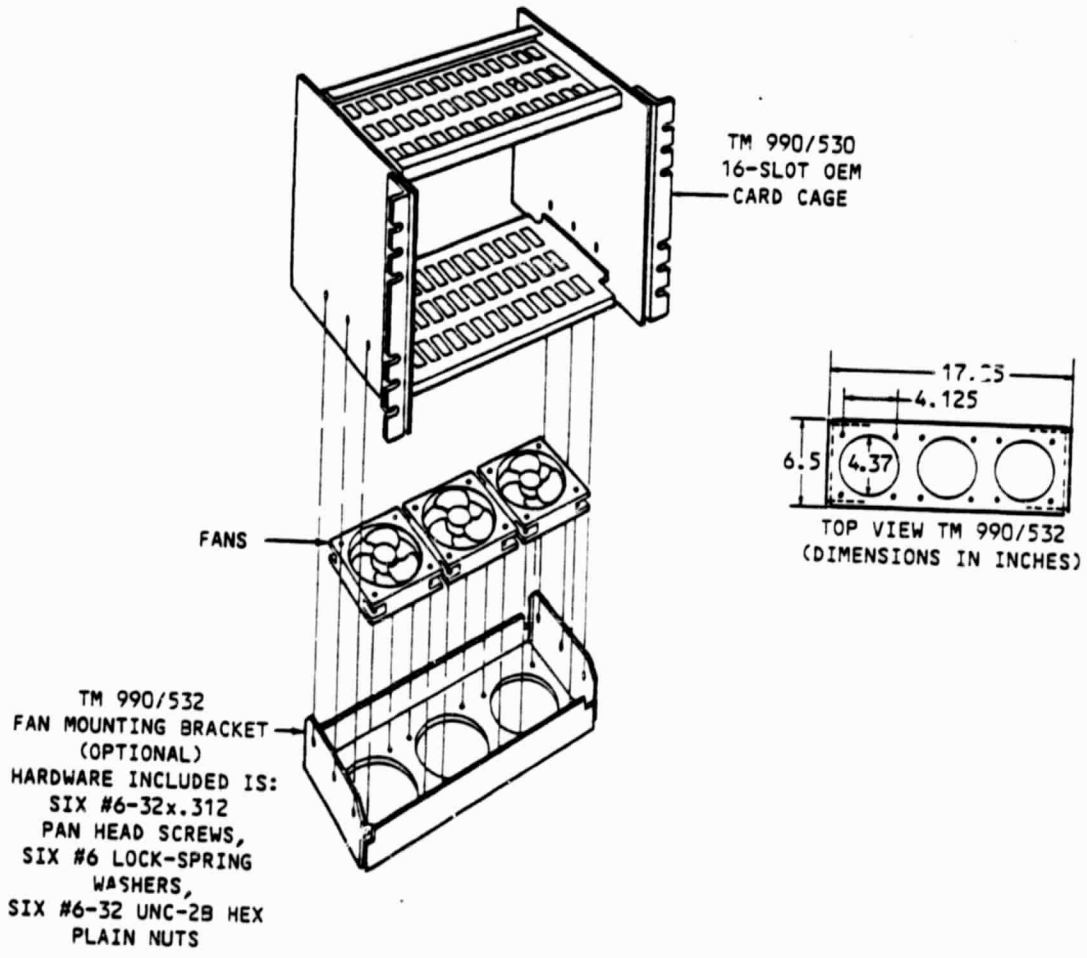


Figure B-6. Cooling Provisions for TM 990/530 Chassis

APPENDIX C

DEFINITIONS OF DIGITAL TERMS

Common digital engineering terms are used in this and subsequent reports on the progress of the controller and data bus hardware and software. These terms are herein defined as applied to this study:

BIT = A binary digit ("0" or "1")

BLOCK = The field of contiguous bits between two synchronization bytes

BYTE = A specific field of 8 contiguous bits.

CHARACTER = A byte representing a teletype hieroglyphic; typically alphanumerical plus punctuation and selected common mathematical symbols.

FLAG = The unique telltale byte "01111110" (serial); marks the start and end of frames.

FRAME = Two flag bytes and the field of bits between them.

MESSAGE = One or more frames; a lenghty message may require several frames for complete transmission.

NIBBLE = A specific field of 4 bits; the first 4 bits or last 4 bits of a byte (least significant nibble, most significant nibble).

WORD = A specific field of two contiguous bytes (16 bits).

APPENDIX D

HEXI-DECIMAL CHARACTER DEFINITIONS

Bit Pattern[1] Most Significant Digit First:					Bit Pattern[2] Most Significant Digit First:			
2^3	2^2	2^1	2^0	Hexi-Decimal	2^0	2^1	2^2	2^3
0	0	0	0	0	0	0	0	0
0	0	0	1	1	1	0	0	0
0	0	1	0	2	0	1	0	0
0	0	1	1	3	1	1	0	0
0	1	0	0	4	0	0	1	0
0	1	0	1	5	1	0	1	0
0	1	1	0	6	0	1	1	0
0	1	1	1	7	1	1	1	0
1	0	0	0	8	0	0	0	1
1	0	0	1	9	1	0	0	1
1	0	1	0	A	0	1	0	1
1	0	1	1	B	1	1	0	1
1	1	0	0	C	0	0	1	1
1	1	0	1	D	1	0	1	1
1	1	1	0	E	0	1	1	1
1	1	1	1	F	1	1	1	1

Notes: (1) Typical video display of processor logic registers (parallel data transfer).

(2) Typical oscilloscope display of data bus bit transmission (serial data transfer).

APPENDIX E

DERIVATION OF MINIMUM LIFE CYCLE COST

To minimize the life cost of the batteries, the extrema of the life cost equation are found for varying depth of discharge (DOD). To minimize the life cost, the mean life must be at its maximum. The mean life of a two-battery system is given by

$$\text{Life}_M = 2\left(\frac{L_1 L_2}{L_1 + L_2}\right)$$

where $L_1 = 10^{b-ax_1}$ and $L_2 = 10^{b-ax_2}$.

Since $x_1 + x_2 = DOD_1 + DOD_2 = K$, the derivative of the mean life is taken with respect to $DOD_1 = X_1$ and set equal to zero.

$$\frac{d}{dx_1}\left(\frac{L_1 L_2}{L_1 + L_2}\right) = \frac{d}{dx_1}\left(\frac{10^{b-ax_1} \cdot 10^{b-ax_2}}{10^{b-ax_1} + 10^{b-ax_2}}\right) = 0$$

$$\frac{d}{dx_1}\left(\frac{10^{2b-aK}}{10^{b-ax_1} + 10^{b-a(K-x_1)}}\right) = 0$$

Note: $$\frac{d}{dx_1}\left(\frac{A \cdot B}{A + B}\right) = \frac{\frac{d}{dx_1}(A + B)\ AB - \frac{dAB}{dx_1}(A + B)}{(A + B)^2}$$

and $$\frac{d}{dx_1}(10^{b-ax_1}) =$$

$$\rightarrow \frac{d}{dx_1}\left(10^{b-ax_1} + 10^{b-a(k-x)_1}\right) = 0$$

$$\rightarrow -a10^{b-ax_1}(\ln 10) + a10^{b-ak+ax_1}(\ln 10) = 0$$

C-3

$$\rightarrow +b - ax_1 = b - ak + ax_1$$

$$\rightarrow x_1 = K/2 \text{ is the extrema point, that is } x_1 = x_2 = K/2$$

This is an extrema point, whether a maximum or minimum remains to be determined. For $x_1 = K/2$

$$\text{Life}_M = 2\left(\frac{L_1 L_2}{L_1 + L_2}\right) = 10^{b-aK/2} = M$$

Letting $x_1 = K/2 + \varepsilon$ where $\varepsilon \rightarrow 0$

$$\text{Life}_M = \frac{2M}{10^{-a\varepsilon} + 10^{a\varepsilon}}$$

Now, is M a minimum or a maximum?

$$M \gtrless \frac{2M}{10^{-a\varepsilon} + 10^{a\varepsilon}} \text{ for } \varepsilon \neq 0$$

$$1 \gtrless \frac{2}{10^{-a\varepsilon} + 10^{a\varepsilon}} \rightarrow 1 > \frac{2}{10^{-a\varepsilon} + 10^{a\varepsilon}}$$

since $10^{a\varepsilon} + 10^{-a\varepsilon}$ is always greater than 2 for $\varepsilon \neq 0$. Therefore, Life_M is a maximum at $x_1 = x_2 = K/2$. Then it follows that by operating the batteries with a balanced depth of discharge, for any given load the life cost is minimized.

APPENDIX F

DERIVATION OF MAXIMUM ONE-WAY BUS PROPAGATION DELAY

Assumptions:

Signal propagation velocities:

Wire -- 0.15 m/ns
Optical Fiber -- 0.19 m/ns

Maximum bus length: less than 120 meters

Calculation of maximum bus propagation time:

Wire -- 120m ÷ 0.15 m/ns = 800 ns
Optical Fiber -- 120m ÷ 0.19 m/ns = 632 ns

At 1.0 Mb/s serial bus data rate, the maximum propagation delays in bit times are:

Wire -- 0.8 bit time
Optical Fiber -- 0.632 bit time

At 10.0 Mb/s serial bus data rate, the maximum propagation delays are:

Wire -- 8 bit times
Optical Fiber -- 6.32 bit times

A 120-meter wire bus operating at 10.0 Mb/s was selected as the limiting case.*

Worst-Case Maximum** Delay for any BIA to Wait for Bus Access (assuming 30 BIAs on bus):

at 1.0 Mb/s serial bus data rate: 0.58644 second.
at 10.0 Mb/s serial bus data rate: 0.058644 second.

Maximum**Transmission Delay for bus Access With No Information Traffic (with 30 BIAs on bus):

at 1.0 Mb/s serial bus data rate: 2.55 milliseconds.
at 10.0 Mb/s serial bus data rate: 255 microseconds.

Figure F-1 illustrates the limitations of bus access time.

---

*Since the clocks of two BIAs could be a maximum of one bit-time relatively asynchronous, the maximum wire bus length should be limited to 100 meters.

**"Maximum" delay means that software transmit request just missed the bus access period.

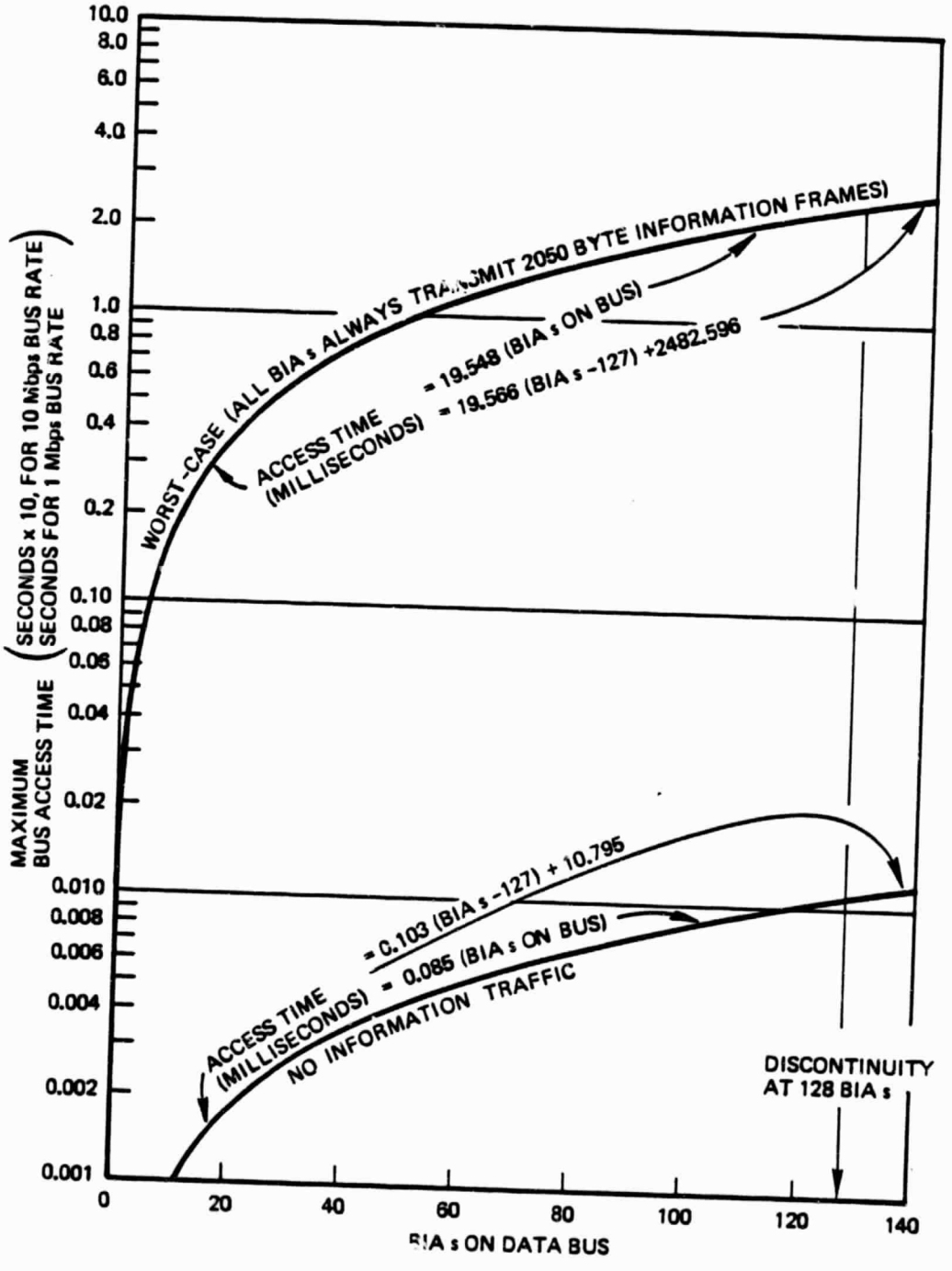


Figure F-1. Data Bus Access Time